

dossier

COMMODORE



**PROGRAMMI
TRUCCHI
DIDATTICA**

COMMODORE



Mensile
di giochi e programmi
per COMPUTER
COMMODORE

Direttore Responsabile
Marco Bindi

Direttore Tecnico:
Gianroberto Negri

Edizioni **CO.STA** srl
via Mazzini, 128
56100 PISA

Registrazione al Tribunale
di Pisa n. 2 del 22/2/84
Spediz. in abbon. post.
gruppo III/70

COMMODORE DOSSIER

il mensile per realizzare la tua Biblioteca
di Software in tre sezioni:

- Giochi
- Programmi personali
- Programmi di utilità

Lire 2000

I numeri arretrati, a lire 4000, possono
essere richiesti senza oneri di spedizione
con pagamento anticipato in contanti,
o tramite assegno o vaglia postale
intestato alla Società editrice.

Stampa:
Rotolito Lombarda Spa (MI)

Distribuzione
Messaggerie Periodici (MI)

COMMODORE 64: SIMBOLI GRAFICI

SIMBOLO LISTATO

[SP2]
[BNC]
[GIU']
[RVS ON]
[HOME]
[ROSS]
[DESTR]
[VRD]
[BLU]
[ST.]
[SHFT +]
[SHFT +]
[COMD -]
[SHFT -]
[P1]
[COMD *]
[F1]
[F2]
[F3]
[F4]
[F5]
[F6]
[F7]
[F8]
[NERO]
[SU]
[REV OFF]
[CLR]
[INST]
[PORP]
[SIN.]
[GIA]
[CYN]
[SHFT SP2]
[COMD ST.]
[SHFT ST.]
[SHFT @]
[DEL]

TASTI DA PREMERE

BARRA SPAZIATRICE
CTRL+2
CRSR(IL PRIMO A SINISTRA)
CTRL+9
HOME
CTRL+3
CRSR(QUELLO A DESTRA)
CTRL+6
CTRL+7
TASTO CON RAFFIGURATA LA STERLINA ()
SHIFT++
SHIFT + IL TASTO +
COMMODORE + IL TASTO -
SHIFT + IL TASTO -
IL TASTO PIGRECO
COMMODORE + *
TASTO DI FUNZIONE F1
TASTO DI FUNZIONE F2
TASTO DI FUNZIONE F3
TASTO DI FUNZIONE F4
TASTO DI FUNZIONE F5
TASTO DI FUNZIONE F6
TASTO DI FUNZIONE F7
TASTO DI FUNZIONE F8
CTRL + 1
SHIFT + CRSR(IL PRIMO A SINISTRA)
CTRL + 0
SHIFT + HOME
SHIFT + DEL
CTRL + 5
SHIFT + CRSR(QUELLO A DESTRA)
CTRL + 8
CTRL + 4
SHFT + BARRA SPAZIATRICE
TASTO COMMODORE + TASTO STERLINA ()
SHIFT + TASTO STERLINA ()
SHIFT + @
DEL

NEL NUMERO DI OTTOBRE DI PERSONAL COMPUTER

Programma/Mese:
Agenda personale per C64 e MSX

Le prove su strada:
Pc 10 - Megastore 2 - 800XL - Automa

E gli speciali:
TUTTOCOMMODORE - TUTTAPPLE - TUTTOMSX
SINCLAIRDOSSIER - TEXASDOSSIER

SPRITE 4

In questa quarta ed ultima parte, ci occuperemo, come promesso, della realizzazione di un gioco, utilizzando la tecnica delle SPRITE e diversi altri trucchi.

Il listato di seguito riportato, si può dividere essenzialmente in 4 parti, ognuna d'esse adibita alla gestione di una parte ben precisa del gioco; vediamole: PARTE PRIMA: istruzioni che vanno dall'istruzione 1 all'istruzione 80. In questa prima parte vengono inizializzate le variabili, create e posizionate le due SPRITE utilizzate nel programma, creata la cornice entro la quale si svolge il gioco, creati i contatori delle VITE del giocatore e del punteggio.

PARTE SECONDA: istruzioni che vanno dall'istruzione 90 all'istruzione 220. In questa seconda parte viene gestito il gioco vero e proprio. Mediante le istruzioni che vanno dall'istruzione 100 all'istruzione 140, viene LETTA la tastiera e gestito il movimento del giocatore. Mediante le istruzioni che vanno dalla 150 alla 220, viene gestito lo spostamento RANDOM (casuale) dell'oggetto che il giocatore deve cercare di non far cadere a terra ed il rilevamento delle eventuali collisioni contro i caratteri o contro la SPRITE rappresentante il giocatore.

PARTE TERZA: istruzioni che vanno dalla 230 alla 270. Mediante questa parte, viene gestita la fine del gioco ed i relativi messaggi conclusivi.

PARTE QUARTA: istruzioni che vanno dalla 280 alla 330. Sono contenuti in questa parte i valori delle due SPRITE utilizzate nel gioco.

Individuate quelle che sono le parti del programma che si occupa della gestione del nostro favoloso gioco, vediamo ora di analizzarle in maniera più dettagliata:

LISTA VARIABILI DEL PROGRAMMA

PROGRAMMA : COMMODORE MIX

VAR. : NUMERO ISTRUZIONE

A\$	:	250	260	
CO	:	20	70	80
I	:	40	70	80
LI	:	20	90	180 220

M	:	5	40	150	170	190	210
F	:	20	50	120	130	140	180
PE	:	100	110	120	130		
Q	:	40					
S	:	90	170	190	200		
SC	:	20					
V	:	20	30	50	140	200	210
X	:	20	50	150	180	210	
X1	:	150	160	170	210		
XD	:	20	160	170	180	190	200
Y	:	20	50	150	180	210	
Y1	:	150	160	180	190	200	210
YD	:	20	160	180	190	200	

1 RESTORE

5 M = 832

10 POKE 53281,0:POKE 53280,9

20 CO=54272:P=150:X=150:Y=190:V=53248
:SC=0:LI=3:XD=INT(RND(1)*9-4):YD=-1

30 POKE V+21,3:POKE 2040,13:POKE 2041,14

40 FOR I=0 TO 126:READ Q: POKE M+I,Q:
NEXT

50 POKE V,P:POKE V+1,200:POKE V+2,X:P
DKEV+3,Y

60 PRINT"[CLR][BNC]SUPER UOMO!!![2 SP
Z][BLU]PUNTI:"

70 FORI=53381TO55406:POKEI,5:POKEI-CO
,86:NEXT:FORI=55381TO56156STEP40

80 POKEI,5:POKEI-CO,86:POKEI+25,5:POK
EI+25-CO,86:NEXT

90 PRINT"[HOME][21 DESTR][BNC]";S;"[5
SPZ]";LI

100 PE=PEEK(197)

110 IF PE<>12ANDPE<>36THEN 150

120 IF PE= 12ANDP > 70THEN P=P-2

130 IF PE= 36ANDP <240THEN P=P+2

140 POKE V,P

150 Y1=Y:X1=X:M=0

160 Y1=Y1+YD:X1=X1+XD

170 IF X1<70 OR X1>240 THEN XD=-XD:S=
S+1:M=1

180 IF Y1>220THENX=P:Y=200:XD=INT(RND
(1)*9-4):YD=-1:LI=LI-1:GOTO 220

190 IF Y1<70 THEN YD=1:XD=INT(RND(1)*
9-4):S=S+1:M=1

200 IF Y1<210 AND PEEK(V+30)<>0 THEN
YD=-1:XD=INT(RND(1)*9-4):S=S+2

ISTRUZIONE 40: in essa troviamo le parole chiave FOR e TO che insieme alla variabile di tipo numerico I, generano un ciclo che va dal numero 0 sino al numero 126 per un totale di 127 volte. Di seguito mediante la parola chiave READ vengono via, via, letti i valori numerici presenti dopo la parola chiave DATA nelle istruzioni che iniziano dalla 280 sino alla 330, caricandoli nella variabile di tipo numerico Q. Sempre nella medesima istruzione, mediante la parola chiave POKE, i valori acquisiti mediante la parola chiave READ e la variabile di tipo numerico Q, vengono caricati in memoria a partire dall'indirizzo 832 (M+1 cioè 832 che è il contenuto della variabile di tipo numerico M, viene sommato al valore della variabile di tipo numerico I, via, via incrementata dalle parole chiave FOR, TO e NEXT) sino all'indirizzo 958. La parola chiave NEXT e la variabile numerica I, chiudono il ciclo.

ISTRUZIONE 50: in essa mediante la prima parola chiave POKE, viene posizionata la SPRITE 0 alla posizione orizzontale 150, essendo questo il valore contenuto nella variabile di tipo numerico P. Il valore 53248, caricato nella variabile di tipo numerico V, rappresenta l'indirizzo del REGISTRO del COMMODORE 64 adibito alla gestione della posizione X (ORIZZONTALE) della SPRITE 0. La seconda parola chiave POKE serve per comunicare la posizione Y (VERTICALE) alla quale vogliamo posizionare sempre la SPRITE 0. Viene utilizzata la medesima tecnica utilizzata in precedenza, infatti per individuare il REGISTRO numero 1, corrispondente all'indirizzo di memoria 53249, viene semplicemente aggiunto il numero 1 al contenuto della variabile di tipo numerico V. Ora poiché nella variabile di tipo numerico V è contenuto il valore numerico 53248, sommando 1 a 53248 si ottiene appunto 53249 che è l'indirizzo del REGISTRO a cui vogliamo accedere. La posizione iniziale è 200. La medesima tecnica viene utilizzata con la terza parola chiave POKE; infatti per individuare il REGISTRO numero 2, corrispondente all'indirizzo di memoria 53250, viene semplicemente aggiunto il numero 2 al contenuto della variabile di tipo numerico V. Ora poiché nella variabile di tipo numerico V è contenuto il valore numerico 53248, sommando 2 a 53248 si ottiene appunto 53250 che è l'indirizzo del REGISTRO a cui vogliamo accedere. La posizione iniziale è rappresentata dal valore contenuto nella variabile di tipo numerico X, ed in questo caso 150.

Anche per la quarta parola chiave POKE, viene utilizzata la medesima tecnica; infatti per individuare il REGISTRO numero 3, corrispondente all'indirizzo di memoria 53251, viene semplicemente aggiunto il numero 3 al contenuto della variabile di tipo numerico V. Ora poiché nella variabile di tipo numerico V è contenuto il valore numerico 53248, sommando 3 a 53248 si ottiene appunto 53251, che è l'indirizzo del REGISTRO a cui vogliamo accedere. La

posizione iniziale è rappresentata dal valore contenuto nella variabile di tipo numerico Y, ed in questo caso 190.

ISTRUZIONE 60: in essa mediante la parola chiave PRINT, viene stampata a video, cambiandone il colore, la costante "SUPER UOMO!!! PUNTI".

ISTRUZIONI 70 ed 80: in esse mediante le parole chiave FOR, TO, NEXT e POKE, viene generata sul video, la cornice di caratteri entro la quale si svolge il gioco. La tecnica utilizzata è quella di caricare il codice del carattere che si desidera utilizzare (86), direttamente nella memoria video ed il codice del colore che deve avere il carattere (5), direttamente nella memoria adibita al colore. La posizione della memoria adibita al video, si ottiene sottraendo alla posizione della memoria adibita alla gestione del colore del carattere in quel momento utilizzato, il valore 54272 contenuto nella variabile di tipo numerico CO.

PARTE SECONDA:

ISTRUZIONE 90: in essa mediante la parola chiave PRINT, viene stampata a video il contenuto delle variabili di tipo numerico S, corrispondente alla variabile adibita alla gestione del punteggio del gioco, ed il contenuto della variabile di tipo numerico LI, corrispondente alla variabile adibita al conteggio delle vite ancora a disposizione del giocatore.

ISTRUZIONE 100: in essa mediante la parola chiave PEEK, viene caricato nella variabile di tipo numerico PE, il codice del tasto che in quel momento è stato premuto. Questo è un trucco da tenere in dovuta considerazione, a causa della estrema velocità e versatilità che esso offre.

ISTRUZIONE 110: in essa, mediante le parole IF, AND e THEN viene controllato se il codice del tasto in quel momento premuto, sia diverso dai codici 12 e 36. In caso affermativo, viene effettuato un salto all'istruzione 150. I codici dei tasti, testati (controllati), corrispondono ai tasti utilizzati dal giocatore per spostarsi, verso destra oppure verso sinistra. Quando nessuno dei due codici è rilevato, vuol significare che il giocatore, o non ha premuto i tasti corretti, oppure non ha spostato la sua SPRITE nelle due direzioni possibili.

ISTRUZIONE 120: in essa mediante le parole chiave IF, AND e THEN viene controllato se è stato premuto il tasto, il cui codice è 12 (spostamento verso sinistra) e viene anche controllato se il giocatore sta cercando di spostare la SPRITE, oltre la cornice che delimita l'area in cui si svolge il gioco. La variabile di tipo numerico P è utilizzata per gestire gli spostamenti orizzontali della SPRITE del giocatore; in caso di spostamenti verso sinistra, viene tolto il valore 2 dal valore corrispondente all'attuale posizione della SPRITE.

ISTRUZIONE 130: in essa mediante le parole chiave IF, AND e THEN viene controllato se è stato premuto il

tasto, il cui codice e' 36 (spostamento verso destra) e viene anche controllato se il giocatore sta cercando di spostare la SPRITE oltre la cornice che delimita l'area in cui si svolge il gioco. La variabile di tipo numerico P e' utilizzata per gestire gli spostamenti orizzontali della SPRITE del giocatore; in caso di spostamenti verso destra, viene aggiunto il valore 2 al valore corrispondente all'attuale posizione della SPRITE.

ISTRUZIONE 140: in essa mediante la prima parola chiave POKE, viene posizionata la SPRITE 0 alla posizione orizzontale il cui valore e' contenuto nella variabile di tipo numerico P. Il valore 53248, caricato nella variabile di tipo numerico V, rappresenta l'indirizzo del REGISTRO del COMMODORE 64 adibito alla gestione della posizione X (ORIZZONTALE) della SPRITE 0.

Le istruzioni che vanno dall'istruzione 150 all'istruzione 210, sono adibite alla gestione degli spostamenti della SPRITE mossa dal computer; vediamo di analizzarle. Una premessa pero': a causa della complessita' delle tecniche utilizzate, non scenderemo molto nella loro analisi, riservando alla descrizione della tecnica utilizzata un prossimo numero della rivista.

ISTRUZIONE 150: in essa viene caricato il valore contenuto nella variabile di tipo numerico Y (190) nella variabile di tipo numerico Y1; viene caricato il valore contenuto nella variabile di tipo numerico X (150) nella variabile di tipo numerico X1; viene azzerata la variabile di tipo numerico M.

ISTRUZIONE 160: in essa viene aggiunto il valore contenuto nella variabile di tipo numerico YD alla variabile di tipo numerico Y1; viene aggiunto il valore contenuto nella variabile di tipo numerico XD alla variabile di tipo numerico X1; il valore contenuto nella variabile di tipo numerico XD e' ottenuto in modo RANDOM (CASUALE) mediante le formule contenute nelle istruzioni: 20, 180, 190, 200.

ISTRUZIONE 170: in essa viene controllato che gli spostamenti orizzontali delle SPRITE 1, controllata dal computer, non oltrepassi il bordo sinistro e destro del campo di gioco.

ISTRUZIONE 180: in essa viene controllato se la posizione verticale della SPRITE 1, controllata dal computer e' maggiore di 220, segno che il giocatore non e' riuscito a farla deviare. In caso affermativo, viene tolta una VITA al giocatore e viene effettuato un salto all'istruzione 220.

ISTRUZIONE 190: in essa viene controllato che la posizione verticale della SPRITE 1, controllata dal computer, sia inferiore a 70. In caso affermativo,

viene caricato il valore 1 nella variabile di tipo numerico YD; viene ricalcolata mediante una formula la nuova posizione orizzontale. La formula fa uso della funzione RANDOM per calcolare il valore della posizione in modo CASUALE.

ISTRUZIONE 200: in essa viene controllato se la posizione verticale della SPRITE 1 controllata dal computer e' minore di 210. In caso affermativo, viene controllato se vi sono state delle collisioni tra la SPRITE 0 del giocatore e la SPRITE 1 del computer. Il controllo viene effettuato controllando se il registro V*30 e' diverso da 0. In caso affermativo, viene caricato il valore -1 nella variabile di tipo numerico YD; viene ricalcolata mediante una formula la nuova posizione orizzontale. La formula fa uso della funzione RANDOM per calcolare il valore della posizione in modo CASUALE.

ISTRUZIONE 210: se i vari controlli fatti nelle precedenti istruzioni sono andati a buon fine (valore 0 caricato nella variabile di tipo numerico M), viene posizionata la SPRITE 1 controllata dal computer.

ISTRUZIONE 220: viene controllato se il giocatore ha ancora delle VITE a disposizione. In caso affermativo, viene effettuato un salto all'istruzione 90.

PARTE TERZA:

ISTRUZIONE 230: mediante essa viene stampata a video la costante di tipo alfanumerico "PECCATO E' CADUTO!!!!!!".

ISTRUZIONE 240: mediante essa viene stampata a video la costante di tipo alfanumerico "GIOCHI ANCORA ?".

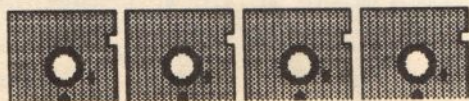
ISTRUZIONE 250: in essa mediante la parola chiave GET e la variabile di tipo alfanumerico A\$, viene prelevato un carattere dalla tastiera, nel caso ovviamente che il giocatore abbia premuto un tasto. Viene poi controllato se esso e' la lettera S. In caso affermativo, viene fatto ripartire il programma dall'inizio, mediante la parola chiave RUN.

ISTRUZIONE 260: in essa mediante le parole chiave IF e THEN viene controllato se nella variabile di tipo alfanumerico A\$, vi sia contenuto una lettera diversa da N. In caso affermativo viene effettuato un salto all'istruzione 250.

ISTRUZIONE 270: mediante essa viene interrotta l'esecuzione del programma.

PARTE QUARTA:

Le istruzioni che vanno dalla 280 sino alla 330, contengono, dopo la parola chiave DATA, i valori della nostra preziosissima SPRITE. Come gia' ribadito, essi vengono letti e caricati in memoria, mediante l'istruzione 40.





CICLI CONDIZIONALI

In questo numero parleremo di CICLI CONDIZIONALI, cioè di quelle istruzioni cicliche che possono essere condizionate mediante un confronto tra il contenuto di una variabile ed un'altra variabile od il confronto tra il contenuto di una variabile ed una costante. Questa possibilità è offerta dalle parole chiave DO, UNTIL e LOOP.

Per comprenderne il corretto utilizzo, vediamo di analizzare un programmino che al posto delle parole chiave DO, UNTIL e LOOP utilizza le parole chiave IF, THEN e GOTO ed in cui il risultato prodotto è identico.

```
10 PRINT "IMMETTI UN NUMERO ";
20 INPUT A
30 IF A = 100 THEN GOTO 100
40 PRINT "NUMERO NON CORRETTO!!"
50 PRINT
60 GOTO 10
100 PRINT "NUMERO CORRETTO!!"
```

```
RUN
IMMETTI UN NUMERO ? 50
NUMERO NON CORRETTO!!
```

```
IMMETTI UN NUMERO ? 100
NUMERO CORRETTO!!
```

Vediamo di analizzarlo, iniziando dall'istruzione 10, in cui mediante la parola chiave PRINT, viene stampata a video la costante di tipo alfanumerico "IMMETTI UN NUMERO " ed il punto e virgola posto di seguito, fa in modo che il punto interrogativo prodotto dalla parola chiave INPUT, dell'istruzione 20, appaia sulla medesima linea.

Continuando nell'analisi, troviamo l'istruzione 20, in cui mediante la parola chiave INPUT, viene fatta fermare l'esecuzione del programma, in attesa che venga immesso un dato ed in questo caso un numero. Osservando l'esempio d'esecuzione riportato di seguito al listato del programma, il primo dato è costituito dal numero 50 che a questo punto si trova caricato nella variabile di tipo numerico A. Nell'istruzione 30 viene controllato, mediante le parole chiave IF e THEN, se il valore numerico contenuto nella variabile di tipo numerico A sia 100. In caso

affermativo, viene effettuato un salto all'istruzione 100. Non è però il nostro caso, poiché nella variabile di tipo numerico A, abbiamo caricato il valore numerico 50. Quindi il salto non viene effettuato e l'esecuzione del programma prosegue dall'istruzione 40, istruzione in cui mediante la parola chiave PRINT viene stampata a video la costante di tipo alfanumerico "NUMERO NON CORRETTO!!". L'istruzione 50, posta di seguito mediante la parola chiave PRINT, crea lo spazio di una linea sul video. L'istruzione 60, mediante la parola chiave GOTO fa effettuare un salto all'istruzione 10. A questo punto ci ritroviamo all'inizio del programma e viene perciò eseguita l'istruzione 10, in cui mediante la parola chiave PRINT, viene stampata a video la costante di tipo alfanumerico "IMMETTI UN NUMERO ", ed il punto e virgola posto di seguito, fa in modo che il punto interrogativo prodotto dalla parola chiave INPUT, dell'istruzione 20, appaia sulla medesima linea. Viene poi eseguita l'istruzione 20, in cui mediante la parola chiave INPUT, viene fatta fermare l'esecuzione del programma, in attesa che venga immesso un dato ed in questo caso un numero. Sempre osservando l'esempio d'esecuzione riportato di seguito al listato del programma, il secondo dato è costituito dal numero 100 che a questo punto si trova caricato nella variabile di tipo numerico A.

Nell'istruzione 30 viene controllato, mediante le parole chiave IF e THEN se il valore numerico contenuto nella variabile di tipo numerico A, sia 100. In caso affermativo, viene effettuato un salto all'istruzione 100.

Essendoci effettivamente il valore numerico 100, nella variabile di tipo numerico A, il salto viene effettuato. A questo punto, ci ritroviamo all'istruzione 100, in cui mediante la parola chiave PRINT viene stampata, a video, la costante di tipo alfanumerico "NUMERO CORRETTO!!".

Il programma appena analizzato è CICLICO (RIPETITIVO) e viene continuamente eseguito, e questo fino a quando non si verifica la condizione testata, mediante l'istruzione 30.

Le istruzioni che vanno dall'istruzione 10 all'istruzione 60, vengono perciò continuamente eseguite, sino a quando nella variabile di tipo numerico A, testata nell'istruzione 30, si trova un valore diverso da 100.

Nel momento in cui il valore contenuto nella variabile di tipo numerico A, diventa 100, le istruzioni che vanno dall'istruzione 40 all'istruzione 60, non vengono

eseguite, mentre viene eseguita l'istruzione 100, istruzione a cui si perviene, mediante la parola chiave GOTO, posta dopo la parola chiave THEN, nell'istruzione 30.

Vediamo ora di trasformare il nostro programma, in modo da poter utilizzare le parole chiave DO, UNTIL e LOOP.

Ecco perciò il programma opportunamente modificato con un esempio d'esecuzione e di seguito la spiegazione dettagliata.

```
10 DO UNTIL A = 100
20 PRINT "IMMETTI UN NUMERO ";
30 INPUT A
40 IF A > 100 THEN PRINT "NUMERO NON CORRETTO!!":PRINT
50 LOOP
100 PRINT "NUMERO CORRETTO!!"
```

```
RUN
IMMETTI UN NUMERO ? 50
NUMERO NON CORRETTO!!
```

```
IMMETTI UN NUMERO ? 100
NUMERO CORRETTO!!
```

Iniziamo dall'istruzione 10 dove troviamo le parole chiave DO e UNTIL, seguite da un confronto: A = 100. Se ricordate questo confronto, nel precedente programma, veniva fatto nell'istruzione 30, mediante le parole chiave IF e THEN. Ora invece viene fatto nella prima istruzione del programma. Quindi la parola chiave DO seguita dalla parola chiave UNTIL, permette di eseguire un controllo PRIMA che venga eseguito il CICLO e questo permette di far eseguire o meno le istruzioni in esso comprese. Questo si traduce in una maggiore velocità del programma in cui vengono utilizzate le parole chiave DO, UNTIL e LOOP. Nel programma precedente la tecnica di far eseguire o meno un gruppo d'istruzione era realizzata, facendo uso delle parole chiave IF e THEN per il confronto, e della parola chiave GOTO per saltare le eventuali istruzioni da non eseguire. Per chi volesse scorrere il listato del programma, troverebbe grosse difficoltà a causa dei continui salti da un'istruzione alla seguente causati dall'utilizzo della parola chiave GOTO. In poche parole: i listati in cui viene utilizzata frequentemente la parola chiave GOTO, sono POCO LEGGIBILI da parte di chi non ha realizzato il programma. E, lato comico, gli stessi sono poco leggibili, a distanza di tempo, anche da chi li ha realizzati. Ritornando all'istruzione 10, il ciclo viene ripetuto fino a quando nella variabile di tipo numerico A, sia presente un valore numerico diverso da 100. Nell'istruzione 20, come nel programma precedente, mediante la parola chiave PRINT, viene stampata a video la costante di tipo alfanumerico "IMMETTI UN NUMERO " ed il punto e virgola posto di seguito, fa in modo che il punto interrogativo prodotto

dalla parola chiave INPUT, dell'istruzione 30, appaia sulla medesima linea.

Nell'istruzione 30, la parola chiave INPUT, fa fermare l'esecuzione del programma, in attesa che venga immesso un dato ed in questo caso un numero. Osservando l'esempio d'esecuzione riportato di seguito al listato del programma, il primo dato è costituito dal numero 50 che a questo punto si trova caricato nella variabile di tipo numerico A.

Nell'istruzione 40 viene controllato, mediante le parole chiave IF e THEN, se il valore numerico contenuto nella variabile di tipo numerico A è diverso da 100. In caso affermativo (è il nostro caso), mediante la parola chiave PRINT, viene stampata a video la costante di tipo alfanumerico "NUMERO NON CORRETTO!!" e la parola chiave PRINT, posta dopo i due punti, produce uno spazio di una linea a video.

La parola chiave LOOP, presente nell'istruzione 50, è quella che fa tornare il programma all'istruzione 10, istruzione dove sono presenti le parole chiave DO e UNTIL nel caso che nella variabile di tipo numerico A, vi sia un valore numerico diverso da 100. È il nostro caso, poiché nella variabile di tipo numerico A, è contenuto il valore numerico 50. Viene perciò eseguito una nuova volta il ciclo le cui istruzioni sono quelle comprese tra le parole chiave DO, UNTIL e la parola chiave LOOP, cioè l'istruzione che vanno dalla 20 alla 40 compresa. Viene eseguita perciò l'istruzione 20, in cui mediante la parola chiave PRINT, viene stampata a video la costante di tipo alfanumerico "IMMETTI UN NUMERO " ed il punto e virgola posto di seguito, fa in modo che il punto interrogativo prodotto dalla parola chiave INPUT, dell'istruzione 30, appaia sulla medesima linea. Nell'istruzione 30, la parola chiave INPUT, fa fermare l'esecuzione del programma, in attesa che venga immesso un dato ed in questo caso un numero. Sempre osservando l'esempio d'esecuzione riportato di seguito al listato del programma il successivo dato è costituito dal numero 100 che a questo punto si trova caricato nella variabile di tipo numerico A.

Nell'istruzione 40 viene controllato, mediante le parole chiave IF e THEN, se il valore numerico contenuto nella variabile di tipo numerico A è diverso da 100. Non è il nostro caso, per cui la seconda parte dell'istruzione 40 non viene eseguita.

La parola chiave LOOP, presente nell'istruzione 50, questa volta, non fa tornare il programma all'istruzione 10, istruzione dove sono presenti le parole chiave DO e UNTIL poiché, il valore contenuto nella variabile di tipo numerico A, è 100.

Viene perciò eseguita l'istruzione 100, in cui mediante la parola chiave PRINT viene stampata, a video, la costante di tipo alfanumerico "NUMERO CORRETTO!!".

Osservando i due programmi, tra il primo, utilizzante le parole chiave IF, THEN e GOTO ed il secondo,

utilizzante le parole chiave DO, UNTIL e LOOP, non sussistono grandi differenze per quello che riguarda il numero d'istruzioni utilizzate per realizzare il ciclo. Anzi a dire il vero, nel secondo esempio di controlli, ne sono stati eseguiti due, contro il singolo controllo fatto nel primo programma. Questo e' vero!... Ma e' anche vero che e' stata eliminata la famigerata parola chiave GOTO e guardando bene, all'istruzione 100, non si perviene mediante un salto!... Ma semplicemente nel caso che il ciclo di controllo che va dall'istruzione 10 all'istruzione 50, abbia permesso la continuazione del programma, dall'istruzione immediatamente successiva a quella in cui e' presente la parola chiave LOOP, e nel nostro caso l'istruzione 100. Se percio' li osserviamo attentamente entrambi, possiamo notare come il programma utilizzando le parole chiave DO, UNTIL e LOOP, sia effettivamente piu' leggibile del programma utilizzando la parola chiave GOTO, unitamente alle parole chiave IF e THEN. In esso non vi sono SALTII ad altre parti di programma che quindi comportano una difficile lettura da parte di chi scorre il listato. Il listato e' FLUIDO non CONTORTO ed incomprensibile. Questa importante caratteristica si traduce in una maggiore velocita', in una maggiore modificabilita'

dello stesso, fattore questo, di estrema importanza, poiche' non e' ancora nato il programma privo di errori e quindi non soggetto a modifiche.

Bisogna ammettere che questa tecnica di programmazione, comporta inizialmente uno sforzo maggiore da parte di chi sta' realizzando il programma, ma i vantaggi che essa porta sono tali da mettere in secondo piano lo sforzo iniziale.

La tecnica di programmazione che utilizza ampiamente le parole chiave del secondo programma (DO, UNTIL e LOOP), ha un nome e si chiama: PROGRAMMAZIONE STRUTTURATA.

A scopo informativo le parole chiave DO, UNTIL e LOOP, vengono utilizzate anche se con alcune variazioni, in linguaggi della potenza del PASCAL, del FORTH od addirittura del nuovo SUPERLINGUAGGIO: il C, la cui potenza e strutturazione e' di tutto rispetto.

Sulle parole chiave DO, UNTIL e LOOP c'e' ancora molto da dire, ma per evitare di complicare ulteriormente le cose, in modo da trovarsi con POCHE IDEE... MA MOLTO BEN CONFUSE!!.. Vi rimandiamo al prossimo numero della rivista, dove potremo maggiormente apprezzare le loro caratteristiche.

A rileggerci sul prossimo numero. A presto... E mi raccomando... Non mancate!...



PRINT

Tra una chiaccherata e l'altra siamo arrivati alla quarta puntata di questa analisi del BASIC e delle sue PAROLE CHIAVE. E' nostro augurio che quanto fino a questo punto e' stato detto Vi sia chiaro. Nel caso che non abbiate compreso qualche argomento, non esitate a scrivere, vedremo di chiarire ulteriormente gli argomenti, attraverso le puntate di questa rubrica. Riprendiamo la parola chiave INPUT utilizzata nella precedente puntata poiche' essa ci servira' per poter analizzare una nuova parola chiave: PRINT. L'istruzione sotto riportata, come si puo' osservare contiene la parola chiave INPUT. Vediamo di analizzarne il funzionamento:

```
10 INPUT A$,A
```

Al lancio del micro-programma viene fatta fermare

l'esecuzione dello stesso, in attesa che vengano caricati dei dati ed in questo caso il nostro indirizzo nella variabile di tipo alfanumerico A\$ ed il nostro numero telefonico nella variabile di tipo numerico reale A.

A questo punto nella variabile di tipo alfanumerico A\$ si trovera' il nostro indirizzo, mentre nella variabile di tipo numerico A reale si trovera' il nostro numero telefonico. Bene a questo punto ci troviamo con dei dati contenuti nelle due variabili. Giunge ora il problema: come utilizzarli nel modo piu' adatto?... Ricorrendo alla parola chiave PRINT che ci permette di stampare a video il contenuto delle due variabili.

Aggiungiamo percio' l'istruzione 20, contenente la parola chiave PRINT al programmino, riportato di seguito e di cui e' riportato anche un esempio d'esecuzione:

```
10 INPUT A$,A
```

20 PRINT A\$;A

RUN

? VIA AURELIA 22,550849

VIA AURELIA 22 550849

Vediamo di analizzarlo accuratamente:

L'istruzione 10 mediante la parola chiave INPUT, fa fermare l'esecuzione del programma in attesa che vengano caricati od immessi dei dati nelle variabili A\$ ed A. Nell'esempio sopra riportato, il nostro indirizzo (VIA AURELIA 22) viene immesso, caricandolo nella variabile di tipo alfanumerico A\$ e separato da una virgola, indicante che il dato da caricare nella variabile A\$ e' terminato, viene immesso il nostro numero telefonico (550849), caricandolo nella variabile di tipo numerico reale A.

A questo punto il nostro indirizzo (VIA AURELIA SUD 22) e' contenuto nella variabile di tipo alfanumerico A\$, mentre il nostro numero di telefono (550849) e' contenuto nella variabile di tipo numerico A.

L'istruzione 20 mediante la parola chiave PRINT provvede a far STAMPARE a video il contenuto della variabile di tipo alfanumerico A\$ e corrispondente al nostro indirizzo (VIA AURELIA SUD 22) ed il contenuto della variabile di tipo numerico reale A corrispondente al nostro numero telefonico (550849). La virgola che si trova tra la variabile di tipo alfanumerico A\$ e la variabile di tipo numerico reale A, fa in modo che il contenuto della seconda variabile sia stampato a video ad una distanza di 16 caratteri dal punto in cui e' stato stampato il contenuto della prima variabile. Ora essendo il nostro indirizzo (VIA AURELIA 22) lungo esattamente 14 caratteri, tra la posizione 14 e la posizione 16 c'e' un carattere di differenza come si puo' notare con facilità dall'esempio d'esecuzione del programma, sopra riportato.

Nel caso si desideri STAMPARE a video il contenuto delle due variabili, uno di seguito all'altro, la virgola posta tra le due variabili dell'istruzione 20 dovra', essere sostituita da un punto e virgola, come si puo' osservare dall'istruzione di seguito riportata:

20 PRINT A\$;A

Il risultato sarebbe il seguente:

VIA AURELIA 22550849

Come si puo' notare e' scomparso lo spazio tra: VIA AURELIA 22 e 550849. A prima vista potrebbe sembrare che abitiamo al numero 22550849!!... Assurdo!... Si rende quindi necessario ricorrere ad un semplice trucco per SPAZIARE il contenuto della prima variabile dal contenuto della seconda, e questo senza ricorrere alla virgola, ma conservando sempre il punto e virgola. Ecco perciò sotto riportato il trucco

con un esempio d'esecuzione e di seguito, eccone la dettagliata spiegazione.

20 PRINT A\$;" ";A

VIA AURELIA 22 550849

Fino a: 20 PRINT A\$; non c'e' nulla di nuovo; quello che e' interessante da osservare e' lo SPAZIO preceduto e seguito dagli apici (si chiamano in questo modo questi : " ") o come vengono anche chiamati: virgolette. Questa e' una COSTANTE. Il perche' si chiami in questo modo e' presto detto: perche' non cambia, non varia, non si modifica mai ed e' quindi, come prima detto: COSTANTE.

Bene, l'aver introdotto una COSTANTE dopo il primo punto e virgola, seguita da un altro punto e virgola, prima dell'ultima variabile ha significato dire al nostro sempre piu' fido MICROCOMPUTER: stampa a video il contenuto della variabile A\$, stampa, di seguito, quanto e' contenuto tra gli apici e nel nostro caso uno spazio ed infine, sempre di seguito, quanto e' contenuto nella variabile A.

Se nella istruzione 20, al posto dello spazio, avessimo messo un altro carattere, questo sarebbe stato stampato nella medesima maniera.

Quindi mediante la parola chiave PRINT e' possibile stampare sia le COSTANTI che, il contenuto delle variabili. Una caratteristica assai comoda non trovate? Vediamo a questo punto di realizzare un piccolo programma utilizzando le parole chiave, sino a questo punto analizzate.

Il programmino e' di seguito riportato, con un esempio d'esecuzione.

```
10 PRINT "IMMETTI IL TUO NOME ";
20 INPUT A$
30 PRINT "IMMETTI IL TUO INDIRIZZO ";
40 INPUT B$
50 PRINT "CIAO ";A$
60 PRINT "CHE ABITI IN ";B$;" ";
70 PRINT "TI SALUTO"
```

RUN

IMMETTI IL TUO NOME ?CYBER

IMMETTI IL TUO INDIRIZZO ?VIA AURELIA SUD 22

CIAO CYBER

CHE ABITI IN VIA AURELIA SUD 22 TI SALUTO

Eccone la spiegazione dettagliata:

ISTRUZIONE 10: in essa mediante la parola chiave PRINT, viene stampata a video la COSTANTE: "IMMETTI IL TUO NOME " ed il punto e virgola posto di seguito fa in modo che il punto interrogativo prodotto dalla parola chiave INPUT, contenuta nell'istruzione 20, venga visualizzato di seguito, sulla medesima riga. Come avete potuto notare, il punto e virgola ha la proprieta' di far stampare o visualizzare sulla

medesima riga qualsiasi cosa; in questo caso si e' trattato del punto interrogativo prodotto dalla parola chiave INPUT. Un'altra interessante cosa da notare e' che: non ha importanza che cio' che si desidera stampare sia posto nella medesima istruzione nella quale si trova il punto e virgola. Infatti nell'esempio ora visto, il punto interrogativo e' prodotto dalla parola chiave INPUT, appartenente all'istruzione successiva alla 10 e per essere precisi, la 20.

ISTRUZIONE 20: in essa mediante la parola chiave INPUT, viene interrotta l'esecuzione del programma, in attesa che venga immesso un DATO per poterlo caricare nella variabile di tipo alfanumerico A\$, ed in questo caso il nostro nome. Una volta caricato esso si trovera' nella variabile di tipo alfanumerico A\$, pronto per poter essere utilizzato in un successivo tempo.

ISTRUZIONE 30: in essa mediante la parola chiave PRINT, viene stampata a video la COSTANTE: "IMMETTI IL TUO INDIRIZZO " ed il punto e virgola posto di seguito fa in modo che il punto interrogativo prodotto dalla parola chiave INPUT, contenuta nell'istruzione 40, venga visualizzato di seguito, sulla medesima riga. Come avete potuto notare, anche in questo caso, il punto e virgola ha la proprieta' di far stampare o visualizzare sulla medesima riga qualsiasi cosa; si tratta del punto interrogativo prodotto dalla parola chiave INPUT.

ISTRUZIONE 40: in essa mediante la parola chiave INPUT, viene interrotta l'esecuzione del programma, in attesa che venga immesso un DATO per poterlo caricare nella variabile di tipo alfanumerico B\$, ed in questo caso il nostro indirizzo. Una volta caricato, esso si trovera' nella variabile di tipo alfanumerico B\$, pronto per poter essere utilizzato in un successivo tempo.

A questo punto ci ritroviamo con il nostro nome ed il nostro indirizzo, rispettivamente caricati nelle variabili di tipo alfanumerico A\$ e B\$. Le istruzioni che seguono ci permetteranno di utilizzarli; vediamo come.

ISTRUZIONE 50: in essa mediante la parola chiave PRINT viene stampata a video la costante "CIAO " ed il punto e virgola posto dopo d'essa, fa in modo che il contenuto della variabile di tipo alfanumerico A\$ venga stampato di seguito. Ora poiche' nella variabile di tipo alfanumerico A\$ e' contenuto il

nostro nome (CYBER nell'esempio riportato), il risultato prodotto da questa istruzione e':

CIAO CYBER

ISTRUZIONE 60: in essa mediante la parola chiave PRINT viene stampata a video la costante "CHE ABITI IN " ed il punto e virgola posto dopo d'essa, fa in modo che il contenuto della variabile di tipo alfanumerico B\$ venga stampato di seguito. Il punto e virgola riportato dopo d'essa fa in modo che venga stampata, sempre di seguito, la costante " ". Anche il punto e virgola posto dopo d'essa, svolge la medesima funzione: fare in modo che la costante presente nell'istruzione 70 venga stampata sempre di seguito.

ISTRUZIONE 70: in essa mediante la parola chiave PRINT viene stampata a video, di seguito alle costanti e variabili contenute nell'istruzione precedente, la costante "TI SALUTO".

Ora poiche' nella variabile di tipo alfanumerico B\$ e' contenuto il nostro indirizzo (VIA AURELIA SUD 22 nell'esempio riportato), il risultato prodotto da queste due istruzioni e':

CHE ABITI IN VIA AURELIA 22 TI SALUTO

Un ultimo piccolo trucco prima di salutarci: quando avete la necessita' di SPAZIARE le stampe a video di costanti e/o variabili di una o piu linee, la parola chiave PRINT, utilizzata cosi' come e', produrra' tale risultato. Ecco un esempio di quanto detto:

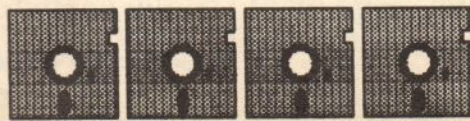
```
10 PRINT "CIAO A TUTTI"  
20 PRINT  
30 PRINT "PER QUESTA PUNTATA"
```

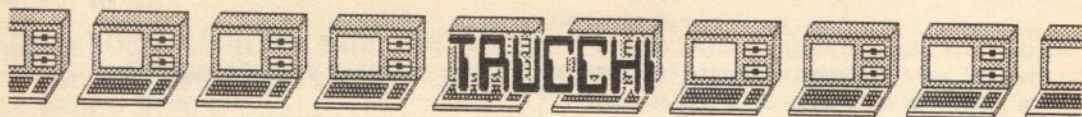
CIAO A TUTTI

PER QUESTA PUNTATA

L'istruzione 20, come si puo' osservare ha prodotto uno spazio di una linea, tra la costante prodotta dall'istruzione 10 e la costante prodotta dall'istruzione 30. Semplice ed efficace!...

Bene anche per questo numero e' giunto il momento di salutarci. A rileggerci sulla prossima puntata.





*A33 ESCAPE

E' noto che nel COMMODORE 64 e nel VIC 20 manca il tasto di ESCAPE. E' una mancanza che si fa particolarmente sentire quanto essi vengono utilizzati, mediante opportune interfacce, con stampanti che non sono quelle standard della COMMODORE. Tali stampanti richiedono infatti, per la gestione dei caratteri da stampare, dei margini, della sottolineatura, dell'interlinea, un insieme di codici di controllo di cui parte prevalente e' il codice corrispondente al tasto ESCAPE. Esiste pero' un trucco per sopperire a tale mancanza: premere il tasto CTRL contemporaneamente al tasto : . Il risultato e' analogo a quello che si sarebbe ottenuto premendo il tasto ESCAPE. Ovviamente il tutto va' racchiuso tra virgolette a causa del particolare modo in cui vengono attivati i codici di controllo nel COMMODORE 64 e nel VIC 20. Ecco un esempio pratico, riferito alla stampante EPSON RX80:

```
10 OPEN 3,4
20 PRINT #3,"CTRL: ":"E";
30 PRINT #3,"PROVA DI STAMPA"
40 CLOSE 3,4
```

Il risultato e' quello di attivare il carattere di stampa che si avvicina maggiormente a quello prodotto da una stampante di tipo a MARGHERITA.

*A34 STOP-CONT

Due comandi poco utilizzati sul COMMODORE 64, sul VIC 20, sul COMMODORE 16 e sul PLUS 4 per il DEBUG dei programmi, sono i comandi STOP e CONT. Essi possono essere utilizzati nel seguente modo per testare l'efficienza dei programmi:

Scegliere il punto dove si prevede un funzionamento anomalo del programma; in quel punto inserire l'istruzione:

```
(NUMERO LINEA) STOP
```

Il programma, quando incontrera', durante l'esecuzione,

questo comando, si interrompera'. A questo punto potra', mediante la parola chiave PRINT, essere visualizzato il contenuto delle varie variabili ed all'occorrenza anche modificato. Effettuati gli opportuni controlli, l'esecuzione potra' essere ripresa dal medesimo punto, dando il seguente comando in modo diretto:

CONT

Utilizzando questa tecnica, potranno essere individuati la maggior parte degli errori presenti in un programma.

*A35 VIC OROLOGIO

Avete l'espansione di memoria dotata di grafica SUPER EXPANDER? In tal caso, ecco un trucco che Vi permettera', premendo semplicemente il tasto di funzione F1, di visualizzare l'orario. Esso sfrutta la variabile di sistema TI\$, entro la quale si trova l'orario corrente. Date perciò il seguente comando in modo diretto:

```
KEY1,"ORE: "+TI$
```

Ogni volta che premerete il tasto di funzione F1, verra' visualizzato l'orario corrente.

*A36 CURSORE

Se volete aumentare la velocita' del cursore sul COMMODORE 64 negli spostamenti verso l'alto dello schermo, date il seguente comando in modo diretto:

```
POKE 56341,1
```

Se volete diminuire la velocita' del cursore, negli spostamenti verso il basso dello schermo, date il seguente comando in modo diretto:

```
POKE 56341,255
```

Se volete avere entrambe le possibilità (cursore verso l'alto veloce e cursore verso il basso rallentato), date il seguente comando in modo diretto:

POKE 650,127

*A37 UTILIZZO DELLA REM

Le istruzioni in cui è contenuta la parola chiave REM, possono essere utilizzate per mettervi, nel caso di programmi tratti dalla rivista, il numero e la pagina di COMMODORE DOSSIER dal quale sono stati tratti. Questo per poter rintracciare velocemente la rivista nel caso di modifiche e/o miglioramenti al programma. Ecco un esempio:

10 REM DIVISIONI - COMMODORE DOSSIER N.5 - PAG. 23

Volendo possono esservi inclusi anche i tasti da utilizzare nel programma. Questo per permettervi a distanza di tempo di conoscerne sempre il corretto utilizzo. Ecco un esempio:

10 REM DIVISIONI - COMMODORE DOSSIER N.5 - PAG. 23
20 REM SPAZIO=PAGINA AVANTI - RETURN=CONFERMA DATO
30 REM F1=RICHIESTA AIUTO

*A38 AGGIORNAMENTO ORARIO

Il COMMODORE 64, possiede come i grossi computer, un orologio interno che può essere utilizzato nell'ambito dei Vostri programmi; basterà aggiungere le seguenti istruzioni:

10 PRINT "IMMETTERE ORA CORRENTE "
20 INPUT "NEL FORMATO HHMMSS : ";OR\$
30 TI\$=OR\$

A questo punto l'orario corrente si trova caricato nella variabile di sistema TI\$, incaricata di trasmettere l'orario all'orologio interno del COMMODORE 64. Mediante la parola chiave PRINT, potrà essere stampato a video, nel punto che più Vi aggrada.

*A39 REM DISCO

Nella realizzazione di programmi in linguaggio macchina

di una certa consistenza, sussiste il problema di dover salvare le varie parti di cui esso è composto. Ogni parte ha però una sua precisa posizione in memoria, posizione che sarebbe utile poter scrivere di lato al nome della parte salvata, così da poter essere visualizzata quando viene richiamata la DIRECTORY del disco. Il trucco che Vi proponiamo, permette di ottenere un tale risultato ed è anche oltremodo semplice da utilizzare. Basta salvare ogni parte nel seguente modo:

SAVE "NOME PARTE (SHIFT+SPAZIO) POSIZIONE DI MEMORIA",8

La pressione contemporanea del tasto SHIFT e del tasto SPAZIO, dopo il nome della parte da salvare ed ovviamente prima della posizione di memoria, provoca, una volta che sia stata caricata la DIRECTORY del disco, il seguente risultato:

"NOME PARTE"POSIZIONE DI MEMORIA

Per fare un esempio: se salviamo su disco una parte di programma in linguaggio macchina nel seguente modo:

SAVE "RANE1 (SHIFT+SPAZIO) LOC.43200",8

Il risultato, una volta richiamata la directory sarà:

"RANE1"LOC.43200

La caratteristica più interessante di questo trucco è che la parte di programma in linguaggio macchina salvata, potrà essere ricaricata, utilizzando SOLAMENTE il nome racchiuso tra apici e non l'intero nome, comprendente anche la posizione di memoria, utilizzato, durante il salvataggio. La parte di nome posta dopo SHIFT+SPAZIO, diventa a tutti gli effetti un commento e viene perciò ignorata dal sistema operativo del disco.

*A40 STAMPA DIRECTORY

Se avete la necessità di stampare il contenuto di un dischetto, da programma, senza intervento manuale, il programma sotto riportato, Ve ne offre la possibilità

10 PRINT"(SHIFT+CLR)(3GIU')LOAD"CHR\$(34)"\$"CHR\$(34)",8"
20 PRINT"(4GIU')OPEN4,4:CMD4:LIST"
30 PRINT"PRINT#4:CLOSE4(HOME)"
40 POKE198,5:FOR J=1TO5:POKE630+J,13:NEXT:END

*A41 ERRORE LETTURA DA NASTRO

Accade che nastri contenenti dei programmi, vengano letti con difficoltà dall'unità a nastro del COMMODORE 64 o del VIC 20. Questo è dovuto per la maggior parte delle volte ad un differente allineamento delle testine tra il registratore con cui è stato registrato il nastro ed il registratore con cui viene effettuata la lettura dello stesso. Si può tentare

lo stesso di far leggere il nastro, intervenendo sulla vitina posta nelle immediate vicinanze della testina di registrazione/lettura del registratore in Vostro possesso. Ruotandola nei due sensi, si può, procedendo per tentativi, orizzontarla di quel tanto che basta a far leggere il nastro.



**CM
64**

**MODULO
LUNARE**

**GG
GG**

Siete ai comandi di un modulo lunare e dovete cercare, una volta atterrati d'evitare che le meteoriti che cadono Vi distruggano. Avete a disposizione un lanciamissili per cercare di colpirle, prima che esse distruggano Voi. I tasti V,B,N permettono di lanciare il missile rispettivamente a: sinistra, sopra di Voi, a destra. Se siete presi dal panico, potete ricorrere ad un'onda d'energia, attivabile mediante il tasto spazio, ma attenzione!... Non ne avete a disposizione molte. Buon divertimento.

SPIEGAZIONE PROGRAMMA

ISTRUZIONI 1-10: cambiano il fondo ed il bordo del video, portandoli a neri. Viene fatto posto in memoria per i valori delle SPRITE.

ISTRUZIONI 19-30: inizializzazione variabili e posizionamento scritta sul bordo superiore dello schermo.

ISTRUZIONI 35-38: generazione della SPRITE del modulo lunare.

ISTRUZIONI 39-45: generazione della SPRITE numero 0.

ISTRUZIONI 49-57: generazione della SPRITE numero 1.

ISTRUZIONI 59-67: generazione della SPRITE numero 2.

ISTRUZIONI 69-75: generazione della SPRITE numero 3.

ISTRUZIONI 79-85: generazione della SPRITE numero 4.

ISTRUZIONI 86-88: generazione della SPRITE della esplosione

ISTRUZIONI 89-98: gestione generazione suoni del gioco ed inizializzazione registri del sintetizzatore interno del COMMODORE 64.

ISTRUZIONI 99-199: gestione generazione superficie del pianeta, visualizzata durante il gioco.

ISTRUZIONI 199-230: gestione razzi del modulo lunare, durante la discesa sulla superficie del pianeta.

ISTRUZIONI 299-500: gestione loop principale del programma.

ISTRUZIONI 999-1100: gestione impatto del modulo lunare con la superficie.

ISTRUZIONI 2999-4020: gestione tastiera (tasti premuti durante lo svolgimento del gioco).

ISTRUZIONI 4499-4540: gestione interruzione del gioco.

ISTRUZIONI 4699-4720: gestione interruzione momentanea del gioco.

ISTRUZIONI 4999-5510: gestione visualizzazione modulo lunare.

ISTRUZIONI 5999-6200: gestione razzi lanciati dal modulo lunare.

ISTRUZIONI 6999-7030: gestione esplosione.

ISTRUZIONI 7499-7510: gestione stato del gioco.

ISTRUZIONI 7999-8060: gestione delle meteoriti.

ISTRUZIONI 9999-10040: gestione messaggio finale in caso di modulo lunare colpito.

ISTRUZIONI 10999-11560: gestione parte finale del gioco e messaggi ad essi inerenti.

ISTRUZIONI 11999-12020: gestione terremoto lunare.

ISTRUZIONI 14999-15100: gestione modulo lunare in caso di missione completata con successo.

ISTRUZIONI 19999-20150: gestione parte introduttiva del gioco.

LISTA VARIABILI DEL PROGRAMMA

PROGRAMMA : MODULO LUNARE

VAR.	NUMERO	ISTRUZIONE
A	36 40 50 60 70 80	
A#	87	
AT	95 11110 11140 11200	
AW	90 95 5000	
B1	140 6200 8000 8010 8020 8050	
BL	11020 11030	
C#	10040 11040	
CL	7506 7510	
CO	15030 15040 15095 15096	
D	140 160 190 200	
DA	1020 1030 12010 12020	
DF	135 1010 7506 7508 7510	
DL	160 190 220	
ER	5080 5090 5110	
FI	315 410 6020 6200	
GI	95 315 410 1010 3000 3510	
GR	6010 7030 11200	
H	67 1050 1060	
H2	1050 1060	
HI	90 230 1000 3505 4010 4520	
KEY	4530 5000 5002 5062 5080 5090	
L	5120 6020 7000 8030 10010 12010	
LO	15020 15070	
MX	3500	
MY	200 305 1000 3505 7005	
N	95 400 4705 4710 4715 11550	
NH	5010 5020 5040	
P	90 230 4520 5000 5005 5020	
PA	5040 15020	
PH	3020 3110 3210 3500 4006 6000	
PL	6020	
PT	3020 3110 3210 3500 3505 4066	
Q	6010 6020	
RDX	3020 3110 3210 3500 6025	
RX	135 7000 11010 11020	
RY	400 1040 1050 3005 3010 3100	
S	3200 4000 4002 4005 11550	
SC#	135 4005 4006 7500	
SC	90 5005	
SC#	90 5005	
SC	135 6200 7500 8020 11000 11030	
SC#	11070 11100 11110 11200 15015	
SC	1040 1060	
SC#	190 200	
SC	3020 3110 3210 3500 6020	
SC#	1010 3020 3110 3210 3500 6010	
SC	6020 7000	
SC#	220 300 320 330	
SC	20 30 36 40 50 60	
SC#	70 80 87 140 5010	
SC	25 11160 11200 11520	
SC#	25 7505 11110 11160 11200 11520	

SP	1000 1040					
SU	90 95 5000					
SX	200 300 305 310 1000 4006					
SY	7010 7020					
T	200 300 310 4006 7010 7020					
V	1000					
VD	20 140 170 180 200 230					
WA	305 310 340 1000 1020 3020					
X	3110 3210 3500 3505 4500 4510					
XC	4530 5010 5020 5040 5065 6000					
XD	6025 7000 7005 7010 7020 7030					
YE	10010 10020 10030 10040 12010 12020					
YF	15000 15010 15020 15030 15040 15060					
YG	15070 15080 15090 15095 15096 15100					
YH	90 95 230 330 1000 1020					
YI	4006 4500 4510 5001 5062 5080					
YJ	6020 7000 7010 7020 8020 10020					
YK	10030 12020 15010					
YL	90 96 230 1010 1040 3505					
YM	4006 4500 4520 5001 5062 5070					
YN	5115 6010 6200 7000 8020 8030					
YO	10010 10040 11005 12010 15010 15060					
YP	15065 15080					
YQ	20 25 30 36 40 50					
YR	60 67 70 80 87 99					
YS	100 135 170 1020 1030 1050					
YT	1060 3500 4010 4510 5020 5025					
YU	5040 5050 5062 5065 5080 5115					
YV	5120 7010 7020 7507 7510 8020					
YW	8030 10010 10020 10030 11110 11120					
YX	11160 11520 11530 12010 12020 15030					
YY	15040 15060 15065 15090 15095 15100					
YZ	20120					
ZA	3020 3110 3210 6000					
ZB	210 300					
ZC	8020 8030					
ZD	220 300					
ZE	4510 4520 4530 5010 5020 5030					
ZF	5040 5065 5500 15060 15070 15095					
ZG	1010 1060 5115 5120 11110 11120					
ZH	11150 11160 11200					

9 REM-----

10 POKES3280,0:POKES3281,0:POKES6,48:
CLR

20 FORX=0T06:S(X)=192+X:NEXTX:V=53248

25 DIMSC\$(10),SC(10):FORX=1T010:SC\$(X)
)=***[PORP1LUNAR [GIA]***]:SC(X)=0:N
EXTX

30 FORX=S(0)*64TDS(6)*64+64:POKEX,0:N
EXTX

35 REM-----

36 FORX=0T050:READA:POKES(1)*64+6+X,A
:NEXTX:POKE2041,S(1)

```

37 DATA64,20,1,16,170,4,11,255,224,21
,85,88,148,0,22,144,0,6,144,0,6,144,0
,6
38 DATA144,0,6,144,0,6,159,255,246,4,
0,16,4,0,16,16,0,4,16,0,4,64,0,1,64,0
,1
39 REM-----
40 FORX=0T07:READA:POKES(0)*64+25+X*3
,A:NEXTX
45 DATA24,62,124,255,254,126,94,8
49 REM-----
50 FORX=0T056:READA:POKES(6)*64+X,A:N
EXTX:POKE2046,S(6)
55 DATA2,0,0,2,0,0,2,0,0,20,0,0,8,16,
0,20,255,0,226,0,128,4,124,64
56 DATA5,66,64,4,66,64,2,66,128,1,67,
0,2,254,128,4,56,64,8,40,32,16,56,16
57 DATA16,40,16,16,56,16,56,40,56
59 REM-----
60 FORX=0T07:READA:POKES(2)*64+25+X*3
,A:NEXTX:POKE2042,S(2)
65 DATA192,224,118,56,24,20,2,0
67 FORX=0T09:READG(X):NEXTX:DATA160,2
27,247,248,98,121,111,100,32,32
69 REM-----
70 FORX=0T07:READA:POKES(3)*64+25+X*3
,A:NEXTX:POKE2043,S(3)
75 DATAB,20,28,28,28,28,42,8
79 REM-----
80 FORX=0T07:READA:POKES(4)*64+25+X*3
,A:NEXTX:POKE2044,S(4)
85 DATA3,7,14,60,24,40,64,0
86 REM-----
87 FORX=0T07:READA:POKES(5)*64+25+X*3
,A:NEXTX:POKE2045,S(5)
88 DATA66,0,4,32,0,34,128,16
89 REM-----
90 VD=54296:WA=54276:AT=54277:H=54273
:LO=54272:PH=54275:PL=54274:SU=54278
95 POKEVD,15:FI=0:A$="":KEY=197:POKEA
T+14,32:POKESU+14,255
98 FORX=WATOWA+14STEP7:POKEX,17:POKEX
,0:NEXT
99 REM-----
100 GOSUB20000:PRINT"[CLR]":FORX=1T01
8:PRINT"[GIU']":NEXT
110 PRINT"[GIA][CMD 0][CMD 1][CMD
0][CMD P][CMD 0][CMD 1][CMD 0][CO
MD 1][CMD P][CMD 1][CMD 0][CMD P]
[CMD 0][CMD P][2 CMD 0][CMD 1][RV
S OFF][CMD Y][RVS OFF][CMD 1][CMD
0][2 CMD P][CMD 0][CMD 0][CMD 0][
CMD P][CMD 0][CMD 0][2 CMD 0][COM
D P][CMD 0][CMD 0][CMD P][CMD 0][
CMD P]";
120 PRINT"[RVS ON][40 SPZ][RVS OFF]";
130 PRINT"[HOME][BNC]MODUOLO LUNARE[2

```

```

SPZJN.5... SEQUENZA DISCESA ";
135 FORX=0T030:POKE1104+RND(1)*600,46
:NEXT:DA=25:PA=3:NH=0:PT=0
140 V=53248:CO=7:POKE2040,S(0):POKEV+
16,0:POKEV+45,3:POKEV+21,127:AW=1
160 GOSUB5000:DF=5:CO=8:GOSUB7500
170 FORX=V+41TOV+43:POKEX,5:NEXT
180 POKEV+44,1
190 DF=DF+2:ROX=11:CO=CO-1
199 REM-----
200 SX=180:SY=10:POKEV,SX:POKEV+1,SY:
POKEV+39,CO:HI=0:ROX=ROX-1:IFROX=0THE
N8000
210 XO=RND(1):IFRND(1)<.5THENXO=-XO
220 YO=.5:S=DF:IFS>20THENS=20
230 POKEWA,129:POKEH,200:POKELO,255:P
OKEVO,1:POKEV+30,0:POKEV+16,0
299 REM-----
300 SX=SX+XO*(S/3):SY=SY+YO*S
305 IFSX>255ANDHI=0THENHI=1:POKEV+16,
PEEK(V+16)+1:8X=0
310 POKEV,SX:POKEV+1,SY:IFSY>190THEN1
000
315 IFFITHENGOSUB6000:IFERTHEN7000
320 S=S+.3:IFS>20THENS=20
330 POKEVD,S/2
340 IF(PEEK(V+30)AND65)=65THEN10000
400 P=PEEK(KEY):IFP<>64THEN3000
410 IFFITHENGOSUB6000:IFERTHEN7000
500 GOTO300
999 REM-----
1000 T=SX+256*HI:SP=INT(T/8)-2:POKEV,
0:POKEV+1,0:POKEV+16,0:POKEVO,15:POKE
H,1
1010 POKEWA+14,0:FI=0:POKERY,0:FORD=1
TORND(1)*4:DA=DA-1:IFDA=0THEN12000
1016 GOSUB7500
1020 FORX=1T010:POKEV+17,26:FORD=1T01
0:NEXT:POKEV+17,27:POKEVO,15-(X/2)
1030 FORD=1T010:NEXTD,X
1040 Q=1784+SP:P=PEEK(Q):IFP=32THENQ=
1824+SP:P=PEEK(Q):POKEWA,0
1050 GR=0:FORX=0T09:IFP=G(X)THENGX=X:
X=10
1060 NEXTX:POKEQ,G(GR+1):NEXTZ
1100 GOTO200
2999 REM-----
3000 IFFITHEN4000
3005 IFP<>31ANDP<>28ANDP<>39THEN4000
3010 IFP<>31THEN3100
3020 MX=164:MY=170:RX=V+4:RY=V+5:XC=-
1:N=4:GOTO3500
3100 IFP<>28THEN3200
3110 MX=164:MY=170:RX=V+6:RY=V+7:XC=0
:N=8:GOTO3500
3200 IFP<>39THEN4000
3210 MX=164:MY=170:RX=V+8:RY=V+9:XC=1

```

```

:N=16
3500 FORX=V+4TOV+9:POKEX,0:NEXT:POKER
X,MX:POKERY,MY:H2=0:N=N+1
3505 POKEWA+14,129:POKEH+14,255-MY:PO
KEV+16,H1
3510 FI=1:GOTO410
4000 IFP=10THEN4500
4002 IFP=41THEN4700
4005 IFP<>60DRPA=0THEN410
4006 PA=PA-1:POKEWA,19:POKEVO,15:MX=S
X:MY=SY
4010 FORX=1TO30:POKE53281,3:POKEH,29:
POKE53281,0:POKEH,200:NEXT:GOSUB7500
4020 GOTO7000
4499 REM-----
4500 POKEV+21,64:POKEWA,129:POKEWA+7,
0:POKEWA+14,0:POKEVO,0
4505 PRINT"[HOME][SHFT v]***[2 SPZ]M
I S S I O N E[2 SPZ]INTERROTTA[5 SPZ]
***"
4510 FORX=1TO15STEP.3:POKEVO,X:YY=YY-
.05*X:POKEV+13,YY:NEXTX
4520 POKEWA,17:POKEWA+7,17:POKEH,32:P
OKEH+7,33:POKELO,YY:POKELO+7,YY+10
4530 YY=YY-.75:POKEV+13,YY:POKEH,255-
YY:POKEH+7,249-YY:IFY>6THEN4530
4540 GOTO11000
4699 REM-----
4700 PRINT"[HOME][NERO][3 SPZ]*** ATT
ESA MODD- 'P' X RIPARTIRE***"
4705 POKE53281,1:IFPEEK(KEY)<>64THEN4
700
4710 IFPEEK(KEY)<>41THEN4710
4715 IFPEEK(KEY)<>64THEN4715
4720 POKE53281,0:PRINT"[HOME][38 SPZ]
":GOTO410
4999 REM-----
5000 POKEAT,32:POKESU,255:POKEAT+7,32
:POKESU+7,255:POKEH,17:POKELO,3
5001 POKEVO,15:POKEWA,17:POKEWA+7,17
5002 POKEH+7,16
5005 POKELO+7,3:POKEPH,8:POKEPL,8
5010 YY=3:L=S(6)*64+13:POKEV+12,168
5020 POKEL,16:POKEL-3,16:FORX=1TO8:YY
=YY+.3:POKEV+13,YY:POKELO,YY:POKELO+7
,YY
5025 NEXTX
5030 GOSUB5500:IFY>175THEN5040
5040 POKEL,0:POKEL-3,0:FORX=1TO8:YY=Y
Y+.3:POKEV+13,YY:POKELO,YY:POKELO+7,Y
Y
5050 NEXTX:GOTO5020
5060 PRINT"[HOME][7 SPZ][SHFT v]ACCEN
SIONE RETTORAZZI ...[8 SPZ]";
5062 POKEWA,129:POKEWA+7,129:POKEH,7:
FORX=15TO1STEP-.08:POKEVO,X:GOSUB5500
5065 YY=YY+.04:POKEV+13,YY:NEXTX

```

```

5067 PRINT"[HOME][VRD][5 SPZ]***SUPER
FICE DEL PIANETA ***[6 SPZ]";
5070 POKEWA,17:POKEWA+7,0
5080 POKEVO,15:FORX=1TO5:PRINT"[HOME]
[2 GIU'][[11 DESTRI][SHFT v]0":POKEH,20
:FORDL=1TO100:NEXT
5090 PRINT"[HOME][2 GIU'][[11 DESTRI][B
NC]0":POKEH,0:FORDL=1TO100:NEXT:NEXT
5100 PRINT"[HOME][2 GIU'][[21 SPZ]"
5110 FORDL=1TO1000:NEXT:PRINT"[HOME][C
NERO]PERICOLO STORMO DI METEORITI IN
ARRIVO[2 SPZ]";
5115 POKEWA,17:FORX=1TO9:POKE53281,X:
FORZ=0TO40STEP.5
5120 POKEH,Z:NEXTZ:NEXTX:POKE53281,0
5200 PRINT"[HOME][40 SPZ]";:RETURN
5500 PRINT"[HOME][2 GIU'][[PORP]ALTITU
DINE[2 SPZ][CYN]";185-INT(VY);"[SIN.]
"
5510 RETURN
5999 REM-----
6000 MX=MX+X*8:IFMX>255THENMX=MX-255
:POKEV+16,PEEK(V+16)+16
6010 MY=MY-8:IFMY<0THENPOKERY,0:FI=0
:POKEWA+14,0:RETURN
6020 POKERX,MX:POKERY,MY:ER=0:POKEH+1
4,255-MY:POKEVO,15
6025 IF(PEEK(V+30)ANDN)=NTHEN6200
6100 RETURN
6200 ER=1:POKEWA+14,0:PT=PT+AW*100:GO
TO7500
6999 REM-----
7000 POKEH,6:POKEVO,15:POKEV+1,0:POKE
RY,0:NH=NH+1:POKEWA,129
7005 IFH1=1THENPOKEV+16,32
7010 FORX=15TO0STEP-.3:POKEV+10,SX:PO
KEV+11,SY:POKEVO,X:POKEV+29,32:POKEV+
23,32
7020 POKEV+10,SX:POKEV+11,SY+4:POKEVO
,X:POKE+29,0:POKEV+23,0:NEXTX
7030 POKEV+11,0:FI=0:GOTO200
7499 REM-----
7500 PRINT"[HOME][21 GIU'][[PORP]PUNTI
:":PT;"[SIN.][3 SPZ][CYN]DISNT.:":PA;
"[SIN.] ";
7505 PRINT"[2 SPZ]P.MS:":SC(1)
7506 C$="[PORP]":IFDA<10THENC$="[SHFT
v]"
7507 X=FRE(0)
7508 IFDA=0THENPRINT"[GIU'][[9 DESTRI][
2 SPZ]":RETURN
7510 PRINT"[GIU'][[GIA]X DANNI :[RVS O
N]";C$;:FORX=1TODA:PRINT"[SHFT P]";:N
EXTX:PRINT"[RVS OFF] ";:RETURN
7999 REM-----
8000 PRINT"[HOME][NERO][6 SPZ]ATTACCO
[3 SPZ]";AW;"[SIN.] COMPLETATO!"

```

```

8010 PRINT"[2 GIU'] [8 SPZ] BONUS[4 SPZ
1100 *";AW
8020 PT=PT+AW*100:POKEVD,15:POKEWA,33
:FORX=0T010:POKE53281,RND(1)*16:FORV=
0T060
8030 POKEH,Y:NEXTY:NEXTX:POKEWA,0:POK
E53281,0
8040 PRINT"[HOME][39 SPZ]"
8045 PRINT"[2 GIU'] [25 SPZ]":GOSUB750
0
8050 AW=AW+1:IFAW<7THEN190
8060 GOTO15000
9999 REM-----
10000 PRINT"[HOME][BNC][5 SPZ]*** MOD
ULO LUNARE KO !! ***[7 SPZ]"
10010 POKEV+21,32:POKEWA,129:POKEH,RN
D(1)*254:FORX=15T00STEP-.1
10020 POKEV+10,170:POKEV+11,180:POKEV
+23,0:POKEV+29,0:POKEVD,X
10030 POKEV+10,166:POKEV+11,174:POKEV
+23,32:POKEV+29,32:POKEVD,0:NEXTX
10040 POKEV+21,0:POKEWA,0:BL=1
10999 REM-----
11000 PRINT"[CLR][BNC] I VOSTRI PUNTI
.. ";PT
11005 POKEWA,0:POKEWA+7,0:POKEWA+14,0
11010 PRINT"[GIU'] A CUI VA SOMMATO IL
BONUS PER IL ";INT((NH/60)/60)
11020 PRINT"[GIU'] PERCENTO DI TIRI AC
CURATI." :B1=INT((NH/60)*5000)
11030 PRINT"[GIU'] BONUS:";B1:PT=PT+B1
11040 IFBL=1THENBL=0:GOTO11100
11050 PRINT"[2 GIU'] BONUS ADDIZZIONAL
E PER GIOCO OK.[6 SPZ]"
11060 PRINT"[GIU'] E MODULO LUNARE INT
ATTO!"
11070 PRINT"[GIU'] BONUS: 5000 PNT." :P
T=PT+5000
11100 PRINT"[2 GIU'] [GIA] PUNTEGGIO FI
NALE E'.....";PT
11110 FORX=0T020:GETA$:NEXTX:Z=0:FORX
=1T010:IFPT>SC(X)THENZ=X:X=11
11120 NEXTX:IFZ=0THEN11500
11130 PRINT"[GIU'] NOME PER TAVOLA PUN
TEGGI[2 SPZ]:";
11140 INPUTA$:IFLEN(A$)>15THENA$=LEFT
$(A$,15)
11150 IFZ=10THEN11200
11160 FORX=9T0ZSTEP-1:SC(X+1)=SC(X):S
C$(X+1)=SC$(X):NEXTX
11200 SC(Z)=PT:SC$(Z)=A$:FI=0
11500 PRINT"[CLR][7 DESTRI][GIA] MIGLIO
RI PUNTEGGI[3 SPZ]"
11510 PRINT"[7 DESTRI][PORP][20 COMD Y
]"
11520 FORX=1T010:PRINT"[GIU'] [CYN]#";

```

```

X;TAB(10);"[BNC]";SC(X);TAB(23);-LGIA
]";SC$(X)
11530 NEXTX:X=FRE(0)
11540 PRINT"[GIU'] [VRD][9 SPZ] 'SPAZIO
'[2 SPZ] PER GIOCAR[3 SPZ]"
11550 P=PEEK(KEY):IFP<>60THEN11550
11560 GOTO1000
11999 REM-----
12000 PRINT"[HOME][6 SPZ]***ATTIVITA'
SISMICA ECC. ***":GOSUB7500
12010 POKEWA,129:POKEH,5:FORX=15T00ST
EP-.07:POKEV+17,26:FORX=1T010:NEXT
12020 POKEV+17,27:FORX=1T010:NEXT:POK
EVD,X:NEXTX:GOTO10000
14999 REM-----
15000 POKEV+21,66:POKEV+28,2:POKEV+37
,7:POKEV+38,2
15005 PRINT"[BNC][HOME] MISSIONE COMPL
ETA. ***** BONUS 5000"
15010 POKEV+23,2:POKEV+29,2:POKEV+40,
4:POKEVD,15:POKEWA,33:POKEWA+7,33
15015 PT=PT+5000:GOSUB7500
15020 POKEH,2:POKEH+7,2:POKELO,1:POKE
LO+7,6:POKEV+3,70
15030 FORX=0T0156STEP.2:POKEV+2,X:CL=
CL+.3:IFCL>15STEPCL=0
15040 POKEV+40,CL:POKEV+37,15-CL:NEXT
X
15060 POKEWA+14,129:FORX=15T00STEP+.1
:YY=YY-.05*(15-X):POKEV+13,YY:POKEV+3
8,X
15065 NEXTX:POKEWA+14,17
15070 YY=YY-.75:POKEV+13,YY:POKEH+14,
200-YY:IFY>77THEN15070
15080 POKEV+38,2:POKEWA+14,0
15090 FORX=156T0226STEP.2:POKEV+2,X
15095 YY=YY-.2:POKEV+13,YY:POKEV+12,X
+12:CL=CL+1:IFCL>15THENCL=0
15096 POKEV+40,CL:POKEV+37,15-CL
15100 POKEV+3,227-X:NEXTX:GOTO11000
19999 REM-----
20000 PRINT"[CLR]*****
"
20010 PRINT"* MODULO LUNARE[4 SPZ]*)"
20020 PRINT"*****"
20030 PRINT
20040 PRINT
20050 PRINT"[GIU'] [BNC]* GIOCO DI STI
LE ARCADE[4 SPZ] COMMODORE-64*"
20060 PRINT"[GIU'] DEFENDI IL MODULO L
UNARE DALLE METEORE"
20070 PRINT"[GIU'] USANDO I RAZZI A TU
A DISPOSIZIONE[5 SPZ]"
20080 PRINT"[GIU'] *****
*****"
20085 PRINT"[GIU'] OGNI 60 METEORE COL

```

```

PITE AVRETE UN[2 SPZ]BONUS"
20090 PRINT"E TORNERETE NELLO SPAZIO!"
"
20100 PRINT"[GIU'] [GIA]CONTROLLI:[CYN
]V,B,N [GIA]FUOCO RAZZI . [CYN]SPAZIO
[3 SPZ]"
20110 PRINT"[GIU'] [GIA]ATTIVA DISINTE
GRATORE TOTALE[CYN]A[GIA] FINE[4 SPZ]

```

```

!!"
20120 FORX=53248TO53248+16:POKEX,0:NE
XT
20130 PRINT"PREMI [CYN]S[GIA] PER PAR
TIRE[2 SPZ]!";
20140 IFPEEK(197)<>13THEN20140
20150 RETURN

```

**CM
64**

NUMERI

**UU
UU**

Questo programma Vi permettera' di controllare se: dati cinque numeri ed un altro numero quale risultato, mediante essi e' possibile pervenire a quest'ultimo, utilizzando le quattro operazioni. Il programma offre, due opzioni: generazione manuale, con i valori ed il numero finale immessi da Voi e generazione automatica dove i valori ed il numero finale vengono generati dal computer. Una volta individuati i dati, potete metterVi tranquillamente a sedere, poiche' possono passare diversi minuti prima che il computer giunga alla soluzione.

SPIEGAZIONE PROGRAMMA

ISTRUZIONI 10-518: gestione menu' iniziale e programma principale.
ISTRUZIONI 900-984: gestione posizionamenti valori.
ISTRUZIONI 1899-2450: gestione addizioni e moltiplicazioni dei valori per cercare di pervenire al numero finale.
ISTRUZIONI 2460-4300: gestione spostamenti del quadrante luminoso, durante la ricerca dei valori.
ISTRUZIONI 4900-5500: gestione routine sonora.
ISTRUZIONI 6000-6320: gestione routine menu' iniziale.
ISTRUZIONI 6330-6600: gestione routine adibita alla generazione automatica dei valori e del numero finale.

LISTA VARIABILI DEL PROGRAMMA

PROGRAMMA : NUMERI

VAR. : NUMERO ISTRUZIONE

A	11	12	15	3200	3220	5100
AI	45	230	240	330	440	450

AX	460	470				
C	46	495	2030	2040	2050	2120
F	917	945	953			
FH	11	13	3205	5205		
FL	11	13	3205	5205		
G\$	3160					
H	11	12	15	3200	3220	5100
I	170					
IC	200	220	230	240	270	280
IX	46	170	250	2010		
JC	210	220	230	240	250	260
K	430	440	450	460	470	480
K9	5000	5010	5205			
L	11	12	57	60	3200	5100
LL	57	962				
M	190	200	210	220	230	240
N\$	60	960	962			
NI	40	45	170	230	240	300
NN	922	2040	2050	2130	2135	2310
O	46	193	495			
S\$	310	320	470			
SI	45	170	200	220	300	350
SE	46	210	270	340	2010	
T	14	38	40	41	920	922
TE	46	420	495	498	915	2035

```

TI : 6350
TI$ : 100 194 512
W : 11 13 15 3205 3220 5205
: 5220
X : 44 45 50 6400
Y : 930
Y$ : 6300 6310 6320
Z$ : 517

```

```

10 REM-----
11 SI=54272:FL=SI:FH=SI+1:W=SI+4:A=SI
+5:H=SI+6:L=SI+24
12 POKE L,15:POKE A,25:POKE H,255
13 POKE FH,50:POKE FL,20:POKE W,33
14 FORT=1T0150:NEXTT
15 POKE W,0:POKE A,0:POKE H,0
19 POKE 53280,7
20 POKE 53281,0
30 POKE 646,1
31 GOSUB 6000
32 PRINT"[CLR]"
34 PRINT" I 6 NUMERI PREGO "
38 FORT=1T06:PRINT
40 INPUTN(T)
41 NEXTT
44 PRINT:INPUT" QUALE E' IL NUMERO DA
TROVARE";X
45 A(0)=X:N(8)=1:S(8)=1
46 SE=7:IX=6:TE=1:Q=1:C=5
47 GOSUB 900
48 POKE 646,7
50 PRINT"[CLR]";TAB(17);"[GIU'] [RVS O
N]" +STR$(X) + " [RVS OFF]"
55 POKE 646,1
57 L=(40-(LL+5))/2
60 PRINT"[3 GIU']";TAB(L);N$(1);N$(2)
;N$(3);N$(4);N$(5);N$(6)
68 REM-----
69 PRINT"[GIU'] [7 SPZ] [26 COMD I]"
70 PRINT " "; "[RVS ON]"; "[38 SPZ]"
72 PRINT"[RVS OFF] [7 SPZ] [26 COMD O]"
73 POKE 646,13:PRINT"[HOME] [RVS ON] [DE
STR] [GIU'] [2 SPZ] [CROND ":PRINT"[HOME]
[RVS ON] [2 GIU'] [2 DESTR] [6 SPZ]"
74 PRINT"[HOME] [RVS ON] [3 GIU'] [3 DES
TR] [4 SPZ]"
75 POKE 646,10:PRINT"[HOME] [RVS ON] [GI
U']";TAB(30);" FASE [3 SPZ]"
76 PRINT"[HOME] [RVS ON] [2 GIU']";TAB(
31);"[7 SPZ]"
78 PRINT"[HOME] [RVS ON] [3 GIU']";TAB(
32);"[5 SPZ]"
80 GOSUB 3072
100 TI$="000000"
170 FORI=1T0IX:S(I)=N(I):NEXTI

```

```

190 M=M+1
191 GOSUB 3072
193 POKE 646,11:PRINT"[HOME] [RVS ON] [2
GIU']";TAB(33);"123":POKE 55408+Q,7:P
OKE 646,13
194 PRINT"[HOME] [RVS ON] [2 GIU'] [2 DE
STR]";TI$
200 I(M)=I(M)+1:IFS(I(M))=0.1 THEN 270
210 J(M)=J(M)+1:IFJ(M)=SETHEN 210
220 IFS(J(M))=0.1 OR J(M)=I(M) THEN 250
230 A(M)=(A(M-1)-N(I(M)))/N(J(M)):IFA
(M)=ABS(INT(A(M))) THEN 310
240 A(M)=(A(M-1)+N(I(M)))/N(J(M)):IFA
(M)=ABS(INT(A(M))) THEN 320
250 IFJ(M)<IX THEN 210
260 J(M)=0
270 IFI(M)<SE THEN 200
280 I(M)=0
290 IFM=1 THEN 495
300 M=M-1:GOSUB 3072:S(I(M))=N(I(M)):S
(J(M))=N(J(M)):GOTO 250
310 S$(M)="+":GOTO 330
320 S$(M)="-"
330 IFA(M)=0 OR A(M)=1 THEN 400
340 IFI(M)=SETHEN 360
350 S(I(M))=0.1
360 IFJ(M)=8 THEN 190
370 S(J(M))=0.1:GOTO 190
400 POKE 646,13
410 PRINT"[HOME] [11 GIU']";TAB(10);"[
RVS ON] IL CONTEGGIO E' OK":POKE 646,1
413 GOSUB 5000
415 PRINT:PRINT
420 IFTE>1 THEN GOSUB 2300
430 FORK=MT01STEP-1
440 IFA(K)=0 OR N(J(K))=0 THEN 480
450 IFA(K)=1 OR N(J(K))=1 THEN 470
460 PRINTTAB(12);A(K);"X";N(J(K));"="
;A(K)*N(J(K))
470 IFN(I(K))<>0 THEN PRINTTAB(12);A(K)
*N(J(K));S$(K);N(I(K));"=";A(K-1)
480 NEXTK
490 PRINT:PRINTTAB(16);"OK 1":GOTO 510
495 TE=TE+1:IFC=5 THEN Q=Q+1
498 IFTE<12 THEN 2000
500 POKE 646,14:PRINT"[HOME] [11 GIU']"
;TAB(1);"[RVS ON] NON CI ARRIVO !!!!!"
502 GOSUB 5000
505 POKE 646,1:PRINT"[3 GIU']";TAB(19)
;"?"
510 M=0:GOSUB 3072
512 POKE 646,13:PRINT"[HOME] [RVS ON] [2
GIU'] [2 DESTR]";TI$
515 PRINT"[16 GIU']":POKE 646,1
517 GETZ$:IFZ$="" THEN 517
518 RUN
900 REM-----

```

```

910 REM[2 SPZ]RICERCA VALORI
912 REM[2 SPZ]IN PIU' E MENO
914 REM-----
915 IFTE>1THEN984
917 F=0
920 FORT=1T06
922 NN(T)=N(T)
925 IFN(T)=>N(T+1)THENGOTO950
930 Y=N(T):N(T)=N(T+1):N(T+1)=Y
945 F=1
950 NEXTT
953 IFF=1THEN917
955 FORT=1T06
960 N$(T)="[RVS ON]" + STR$(N(T)) + " [RV
S OFF] "
962 LL=LL+LEN(N$(T))-3
964 NEXTT
984 RETURN
1899 REM
1900 REM-----
1905 REM
1910 REM ADDIZIONE E MOLTIPLICAZIONE
1915 REM DEI VALORI
2000 REM-----
2005 REM
2010 IX=5:SE=6:M=0
2030 C=C+1:IFC=6THENC=1
2035 IFTE<7THEN2040
2037 GOTO2050
2040 N(C)=NN(C)+NN(C+1):GOTO2100
2050 N(C)=NN(C)*NN(C+1)
2060 REM-----
2100 FORT=1T05
2120 IFT=CTHEN2140
2130 N(T)=NN(T+1)
2135 IFT<CTHENN(T)=NN(T)
2140 NEXTT
2145 N(6)=0
2200 GOTO170
2250 REM-----
2300 IFTE>6THEN2320
2310 PRINTTAB(12);NN(C);"+";NN(C+1);"
=";N(C):GOTO2450
2320 PRINTTAB(12);NN(C);"X";NN(C+1);"
=";N(C)
2450 RETURN
2460 REM
2462 REM-----
2466 REM
2468 REM[4 SPZ]QUADRANTE LUMINOSO
2470 REM
2480 REM-----
2482 REM
3072 POKE646,14:PRINT"[HOME][RVS ON][
8 GIU']";TAB(8);"[4 SPZ]"
3074 POKE646,5:PRINT"[HOME][RVS ON][8
GIU']";TAB(12);"[4 SPZ]"

```

```

3076 POKE646,8:PRINT"[HOME][RVS ON][8
GIU']";TAB(16);"[4 SPZ]"
3078 POKE646,4:PRINT"[HOME][RVS ON][8
GIU']";TAB(20);"[4 SPZ]"
3080 POKE646,7:PRINT"[HOME][RVS ON][8
GIU']";TAB(24);"[4 SPZ]"
3082 POKE646,2:PRINT"[HOME][RVS ON][8
GIU']";TAB(28);"[4 SPZ]"
3148 POKE646,1
3150 IFM<>0THENPRINT"[HOME][RVS ON][8
GIU']";TAB(4+(M*4));"[4 SPZ]"
3160 GETG$:IFG$<>" "THENRUN
3200 POKEL,15:POKEA,25:POKEH,0
3205 POKEFH,(20*M):POKEFL,20:POKEW,33
3210 FORT=1T025:NEXTT
3220 POKEW,0:POKEA,0:POKEH,0
4300 RETURN
4900 REM
4905 REM-----
4910 REM GESTIONE SUONI
4915 REM
4920 REM-----
4925 REM
5000 K9=50
5010 IFPEEK(646)=14THENK9=4
5100 POKEL,15:POKEA,25:POKEH,255
5205 POKEFH,K9:POKEFL,0:POKEW,33
5210 FORT=1T01000:NEXTT
5220 POKEW,0:POKEA,0:POKEH,0
5500 RETURN
6000 REM
6002 REM-----
6004 REM[3 SPZ]GESTIONE MENU'
6010 REM
6015 REM-----
6020 REM
6080 PRINT"[CLR][3 GIU']";TAB(15);"SC
EGLIERE: "
6100 PRINT"[4 GIU']"
6105 PRINT"[DESTR]IMMISSIONE MANUALE
DEI NUMERI.....1"
6200 PRINT:PRINT:PRINT
6210 PRINT"[DESTR]IMMISSIONE AUTOMATI
CA DEI NUMERI....2"
6300 GETY$:IFY$=""THEN6300
6310 IFY$="1"THENRETURN
6320 IFY$<>"2"THEN6300
6330 REM
6333 REM-----
6334 REM NUMERAZIONE AUTOMATICA
6335 REM
6336 REM-----
6337 REM
6350 AX=RND(-T1)
6400 X=INT(RND(1)*899)+101
6500 FORT=1T06
6520 N(T)=INT(RND(1)*14)+1

```

6530 IFN(T)>10THENN(T)=((N(T)-10)*25)
6540 NEXTT

6550 GOTO45
6600 END

CM TOMBOLA GG 64 GG

Vi piace dilettarvi nel gioco della TOMBOLA?... E' un gioco che e' tornato prepotentemente di moda, grazie ai vari rotocalchi. Al lancio del programma potrete scegliere tra quattro varianti di gioco:

1) il lancio della biglia e l'individuazione della posizione del numero e' affidata interamente al computer.

2) il lancio della biglia e' fatto MANUALMENTE, premendo un tasto, mentre l'individuazione del numero sulle cartelle e' affidata al computer.

3) il lancio della biglia e' fatto MANUALMENTE, premendo un tasto, mentre l'individuazione del numero sulle cartelle viene effettuata, mediante il joystick, connesso alla PORTA 2 ed il pulsante di fuoco del medesimo.

4) il lancio della biglia e' fatto dal computer, mentre l'individuazione del numero sulle cartelle viene effettuata, mediante il joystick, connesso alla PORTA 2 ed il pulsante di fuoco del medesimo.

La terza e quarta versione, fanno uso della casella bianca, posizionata al centro delle cartelle per segnalare il corretto posizionamento del cursore, sul numero uscito.

L'istruzioni con piu' di 80 caratteri, vanno digitate, ricorrendo al trucco che si avvale delle abbreviazioni delle parole chiave. Buon divertimento.

DESCRIZIONE PROGRAMMA

ISTRUZIONI 1-23: Inizializzazione variabili del programma e visualizzazione videata iniziale del gioco.

ISTRUZIONI 24-25: Lettura e caricamento in memoria della parte del programma in linguaggio macchina.

ISTRUZIONI 26-33: Forzatura del colore nero nelle locazioni di memoria adibite alla gestione del colore dei caratteri.

ISTRUZIONI 34-44: Gestione parte iniziale del gioco, mediante richiamo delle varie routine.

ISTRUZIONI 45-49: Loop principale del programma.

INIZIO ROUTINE

ISTRUZIONI 50-69: gestione costruzione cartelle della tombola.

ISTRUZIONI 70-94: estrazione del numero e sua gestione.

ISTRUZIONI 95-102: Gestione selezione del prossimo numero della biglia (scelta della lettera e del

numero), controllo che non sia gia' stato estratto.

ISTRUZIONI 103-158: gestione della visualizzazione del numero sulla biglia posta centralmente al video; gestione del suono, del colore e del posizionamento delle sprite utilizzate nel gioco.

ISTRUZIONI 159-173: gestione del movimento e del suono delle biglie utilizzate per l'estrazione del numero.

ISTRUZIONI 174-179: gestione del suono di campanelli.

ISTRUZIONI 180-185: gestione del suono del bongo.

ISTRUZIONI 186-203: gestione del vincitore.

ISTRUZIONI 204-212: sottoroutine di gestione del vincitore e inno finale di vittoria.

ISTRUZIONI 213-246: gestione dei movimenti effettuati mediante il joystick.

LISTA VARIABILI DEL PROGRAMMA

PROGRAMMA : TOMBOLA

VAR. : NUMERO ISTRUZIONE

A#	11	12	22	48	52	53
	54	55	57	58	59	131
A1	52	53	54	55	58	187
	189	192	195	199		
A2	52	53	54	55	57	58
	187	189	192	195	199	
A3	52	53	54	55	57	187
	189	192	195	199		
A4	52	53	54	55	58	187
	189	192	195	199		
AU	12	19	20	47	114	
B0	237	238				
B1	237	238				
BQ	110	130	205	211		
CC	29	41	75	80	82	113
CC	227	228	230	233		
CD	41	73	75	80	82	87
	93	111	113	117	126	199
	214	228	229			
CL	72	74	75	77	80	82
	87	92				
CM	214	215	216	227	232	233
	237	240	244			

CO	29	30	107	148		
CR	112	113	117	125	190	191
	194					
D0	2	217	218	224		
D1	2	218				
D2	2					
DU	29	157	209			
F	33	37	40	214		
FB	223	225	226			
FH	29	157	208			
FL	29	157	208			
FR	228	229	230			
HB	27	28				
I	25	27	37	40	63	67
	75	135	136	141	142	143
	145	146	148	149	150	152
	153	154	157	208	209	243
	245					
J	79	80	81	141	142	161
	162	165				
J0	219	225	227			
J1	220	225	227			
J2	221	225	227			
J3	222	225	227			
KB	217	224				
LC	31	106				
LI	43					
LN	85	86	87			
M1\$	13	18				
M2\$	14	18				
M3\$	15	17				
M4\$	16	20				
N\$	84	85	88			
N1	29	72	100	101		
N1	104	105	106	121	238	
N2	104	106	120	238		
NC	98	100	101	106	107	113
	117	128	129	159	163	188
	191	194				
NM	98	104	113			
NN	72	109	115	127	241	
NU	98	100	101			
OC	215	232	240	244		
P1	117	118	119	120	121	
PD	218	219	220	221	222	223
PN	88	89				
Q	86	87	88	90	205	211
RN	76	81	82	87	91	
S	32	87	117	199	228	229
SD	141	142	143			
SD	29					
SN	87	89				
SS	2	33	40	118	119	170
	171	172	176	177	178	182
	183	184	202	228	229	237
T	209					
T0\$	51	61	65			

T1\$	52	62	66			
T2\$	53	63	64	67	68	
T3\$	54	63	67			
T4\$	55	64	68			
TT	178	184				
V	2	107	129	132	144	148
	152	153	155	163		
W	202	205	211			
WI	188	190	199			
WJ	198	199	202	203	205	211
WM	199	200	202			
WP	200	201				
XZ	25					
X	29	149	151	152	163	
X0	229	230				
X1	229	230				
Y	162	163	164			
Y	29	128	129	150	151	153
YM	160	162	165			
Z	77	80	82	84		

```

1 POKE53281,1:POKE53280,1
2 SS=54272:V=53248:D0=56323:D1=56320:
D2=56321:POKEV+21,0
3 PRINTCHR$(147),CHR$(17),CHR$(17)
4 PRINTSPC(14);CHR$(18);CHR$(31);"TIP
I DI GIOCO":PRINT
5 PRINTSPC(14);CHR$(28);"*****
*":PRINT
6 PRINTSPC(4);CHR$(144);"1- BOCCIA AU
TOM. - SPOST.AUTOMATICO"
7 PRINTSPC(4);"2- BOCCIA MANUALE[2 SP
Z]-SPOST.AUTOMATICO"
8 PRINTSPC(4);"3- BOCCIA MANUALE[2 SP
Z]- SPOST.MANUALE"
9 PRINTSPC(4);"4- BOCCIA AUTOMAT. - S
POST.MANUALE"
10 PRINT:PRINT"[5 SPZ]IMMETERE SCELTA
[4 SPZ]:"
11 GETA$:IFA$=""THEN11
12 AU=VAL(A$)-1:IFAU<0ORAU>3THEN3
13 M1$="-CAMPANELLO X NUMERO IN CARTE
LLE"
14 M2$="-GONG PER NUMERO NON IN CARTE
LLE "
15 M3$="-UN TASTO X LANC.LA BOCCIA"
16 M4$="-IL JOYSTICK (PORTA 2)-FUOCO
X POSIZ."
17 PRINTCHR$(147),CHR$(17),CHR$(17)
18 PRINTSPC(2);M1$:PRINTSPC(2);M2$
19 IFAU=1ORAU=2THENPRINTSPC(2);M3$
20 IFAU=2ORAU=3THENPRINTSPC(2);M4$
21 PRINT:PRINTSPC(10);"UN TASTO PER P
ARTIRE"

```

```

22 GETA$: IFA$="" THEN 22
23 PRINT CHR$(147)
24 REM*** -----
----- ***
25 FOR I=0 TO 11: READ X%: POKE 28672+I, X%: NEXT I
26 REM*** -----
----- ***
27 FOR I=0 TO 3: HB=216+I
28 POKE 251,0: POKE 252,HB: POKE 253,0: POKE 254,0: SYS 28672: NEXT I
29 DIM SD(4,62), CD(5), X(5), Y(5), FL(21), FH(21), DU(21), C(3,4,4), N(4,14)
30 CD(0)=0: CD(1)=14: CD(2)=4: CD(3)=2: CD(4)=7: CD(5)=0
31 L(0)=130: L(1)=137: L(2)=142: L(3)=135: L(4)=143
32 S(0)=1104: S(1)=1127: S(2)=1624: S(3)=1647
33 F(0)=1271+SS: F(1)=1294+SS: F(2)=1791+SS: F(3)=1814+SS
34 GOSUB 50
35 GOSUB 133:
36 REM*** -----
----- *
37 FOR I=0 TO 3: POKE F(I), 1: POKE F(I)+1, 1: NEXT I
38 GOSUB 70
39 REM*** -----
----- ***
40 FOR I=0 TO 3: POKE F(I)-SS, 160: POKE F(I)+1-SS, 160: POKE F(I), 8: POKE F(I)+1, 8: NEXT I
41 FOR CD=0 TO 3: C(CD,2,2)=0: NEXT CD
42 REM*** -----
----- ***
43 FOR LI=0 TO 6: POKE (1520+LI), 64: NEXT LI
44 REM*** -----
----- ***
45 GOSUB 96
46 GOSUB 103
47 IFAU=0 OR AU=3 THEN 45
48 GETA$: IFA$="" THEN 48
49 GOTO 45
50 REM -----
----- ***
51 T0$=" B[2 SPZ]I[2 SPZ]N[2 SPZ]G[2 SPZ]O "
52 A1=176: A2=96: A3=178: A4=174: GOSUB 57: T1$=A$
53 A1=98: A2=32: A3=98: A4=98: GOSUB 57: T2$=A$
54 A1=171: A2=96: A3=123: A4=179: GOSUB 57: T3$=A$
55 A1=173: A2=96: A3=177: A4=189: GOSUB 57: T4$=A$
56 GOTO 61
57 A$=CHR$(A2)+CHR$(A3)+CHR$(A2)
58 A$=CHR$(A1)+CHR$(A2)+A$+A$+A$+A$+CHR$(A2)+CHR$(A4)

```

```

59 A$=A$+"[7 SPZ]" + A$
60 RETURN
61 PRINT CHR$(18) CHR$(31) TO $CHR$(146) S PC(8) CHR$(18) CHR$(31) TO $CHR$(146)
62 PRINT CHR$(144); T1$
63 FOR I=1 TO 4: PRINT T2$: PRINT T3$: NEXT I
64 PRINT T2$: PRINT T4$: PRINT
65 PRINT CHR$(18) CHR$(31) TO $CHR$(146) S PC(8) CHR$(18) CHR$(31) TO $CHR$(146)
66 PRINT CHR$(144); T1$
67 FOR I=1 TO 4: PRINT T2$: PRINT T3$: NEXT I
68 PRINT T2$: PRINT T4$;
69 RETURN
70 REM*** -----
----- ***
71 REM*** -----
----- ***
72 FOR CL=0 TO 4: FOR NN=0 TO 14: N(CL,NN)=0: NEXT NN: NEXT CL
73 CD=0
74 CL=0:
75 FOR I=0 TO 4: C(CD,CL,I)=0: NEXT I
76 RN=0
77 Z=INT(RND(0)*15)+1+15*CL
78 REM*** -----
----- ***
79 J=0
80 IFC(CD,CL,J)=Z THEN 77
81 IF J<RN THEN J=J+1: GOTO 80
82 C(CD,CL,RN)=Z
83 REM*** -----
----- ***
84 N$=STR$(Z)
85 LN=LEN(N$)
86 FOR Q=1 TO LN-1
87 SN=S(CD)+CL*3+RN*80+Q+(3-LN)
88 PN=ASC(MID$(N$,Q+1,1))
89 POKE SN,PN
90 NEXT Q
91 IFRN<4 THEN RN=RN+1: GOTO 77
92 IF CL<4 THEN CL=CL+1: GOTO 75
93 IF CD<3 THEN CD=CD+1: GOTO 74
94 RETURN
95 REM *** -----
----- ***
96 GOSUB 159
97 REM*** -----
----- ***
98 NU=INT(RND(0)*15): NM=NU+1+NC*15
99 REM*** -----
----- **
100 IF N(NC,NU)=1 THEN 97
101 N(NC,NU)=1
102 RETURN
103 REM*** -----
----- ***
104 N1=INT(NM/10): N2=NM-N1*10
105 IF N1=0 THEN N1=-16
106 POKE 1282,L(NC): POKE 1283,N1+176: POKE 1284,N2+176

```

```

107 POKEV+44,CO(NC)
108 REM*** -----
--- ***
109 NN=0
110 BO=0
111 CD=0
112 CR=0
113 IFC(CD,NC,CR)<>NMTHEN125
114 IFAU>1THEN213
115 NN=1
116 REM*** -----
----- ***
117 P1=S(CD)+CR*BO+NC*3+1
118 POKEP1+SS,8
119 POKEP1+1+SS,8
120 POKEP1+1,N2+176
121 POKEP1,N1+176
122 GOSUB175:
123 GOSUB186:
124 GOTO126
125 IFCR<4THENCRCR=CR+1:GOTO113
126 IFCD<3THENCDCD=CD+1:GOTO112
127 IFNN=0THENGOSUB181
128 Y(NC)=230
129 POKEV+1+NC*2,Y(NC):
130 IFBO=0THENRETURN
131 GETA$:IFA$=""THEN131
132 POKEV+21,0:RESTORE:CLR:GOTO1
133 REM*** ----- ***
134 REM*** ----- **
*
135 FORI=0TO3
136 POKE251,I*64:POKE252,57:POKE253,6
3:POKE254,00:SYS28672:NEXTI
137 POKE251,00:POKE252,58:POKE253,63:
POKE254,00:SYS28672
138 REM*** ----- ***
139 POKE251,00:POKE252,212:POKE253,24
:POKE254,00
140 REM*** -----
- ***
141 FORI=0TO4:FORJ=22TO46:READSD
142 POKE(14592+I*64+J),SD:NEXT:
NEXT
143 FORI=0TO63:READSD:POKE14912+I,SD:
NEXT
144 POKEV+21,63:
145 FORI=0TO5
146 POKE2040+I,228+I
147 REM*** ----- ***
148 POKEV+39+I,CO(I):
149 X(I)=145+I*12
150 Y(I)=230
151 X(5)=156:Y(5)=83
152 POKEV+I*2,X(I)
153 POKEV+1+I*2,Y(I)
154 NEXTI
155 POKEV+23,32:POKEV+29,32

```

```

156 REM*** ----- ***
157 FORI=0TO21:READFL(I),FH(I),DU(I):
NEXT
158 RETURN
159 NC=INT(RND(1)*5)
160 YM=INT(RND(0)*13):IFYM<5THENYM=5:
161 J=0
162 Y=J^2-2*YM*J+230
163 POKE(V+NC*2),X(NC):POKE(V+1+NC*2)
,INT(Y)
164 IFY<=132THENRETURN:
165 IFJ<2*YMTHENJ=J+1:GOTO162
166 GOSUB169
167 GOTO159
168 REM*** ----- ***
169 SYS28672
170 POKESS,255:POKESS+1,93:POKESS+5,2
171 POKESS+22,104:POKESS+23,1:POKESS+
24,79:POKESS+4,129
172 POKESS+24,0
173 RETURN
174 REM*** ----- ***
175 SYS28672
176 POKESS+1,255:POKESS+5,9:POKESS+15
,225:POKESS+24,15
177 POKESS+4,21
178 FORTT=1TO300:NEXT:POKESS+4,20
179 RETURN
180 REM*** ----- ***
181 SYS28672
182 POKESS+1,55:POKESS+5,9:POKESS+15,
25:POKESS+24,15
183 POKESS+4,21
184 FORTT=1TO300:NEXT:POKESS+4,20:POK
ESS+24,0
185 RETURN
186 REM*** -----
-- ***
187 A1=3:A2=80:A3=0:A4=0
188 WI=NC:GOSUB198
189 A1=80:A2=3:A3=0:A4=0
190 WI=CR:GOSUB198
191 IFNC<>CRTHEN194
192 A1=0:A2=3:A3=80:A4=0
193 GOSUB198
194 IFNC<>(4-CR)THENRETURN:
195 A1=0:A2=3:A3=0:A4=80
196 GOSUB198
197 RETURN
198 WJ=0
199 WM=S(CD)+WI*A1+WJ*A2+WJ*A3+(4-WJ)
*A4+1
200 WP=PEEK(WM)
201 IFWP<128THENRETURN
202 W(WJ)=WM+SS
203 WJ=WJ+1:IFWJ<=4THEN199
204 REM*** ----- ***

```

```

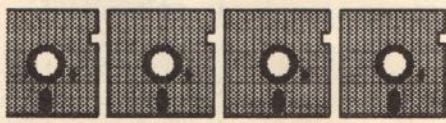
205 FOR WJ=0TO4:FORQ=0TO1:POKEW(WJ)+Q
,7:NEXT:NEXT:B0=1
206 REM*** ----- ***
207 POKE54296,15:POKE54277,10:POKE542
78,10
208 FORI=0TO21:POKE54272,FL(I):POKE54
273,FL(I):POKE54276,33
209 FORT=0TODU(I)*4:NEXT:POKE54276,0:
NEXT
210 REM*** ----- ***
211 FORWJ=0TO4:FORQ=0TO1:POKEW(WJ)+Q,
6:NEXT:NEXT:B0=1
212 RETURN
213 REM -----***
**
214 CM=F(CD)
215 OC=PEEK(CM)
216 POKECM,5:POKECM+1,5
217 KB=PEEK(D0)
218 POKED0,255:PD=PEEK(D1)
219 J0=-((PDAND8)=0)
220 J1=-((PDAND2)=0)
221 J2=-((PDAND4)=0)
222 J3=-((PDAND1)=0)
223 FB=-((PDAND16)=0)
224 POKED0,KB
225 IF(J0+J1+J2+J3+FB)=0THEN242
226 IFFB=1THEN235
227 CC=CM+(J0-J2)*3+(J1-J3)*80
228 FR=INT((CC-S(CD)-SS)/40)
229 X0=S(CD)+SS+FR*40:X1=X0+14
230 IFCC>X1ORCC<X0ORFR>80RFR<0THEN242
231 REM ----- *
*****
232 POKECM,OC:POKECM+1,OC
233 CM=CC
234 GOTO215
235 REM -----
----- *****
236 REM
237 B0=PEEK(CM-SS)-48:B1=PEEK(CM-SS+1
)-48
238 IFB0=N1ANDB1=N2ORB0=N1+128ANDB1=N
1+128THENGOTO115

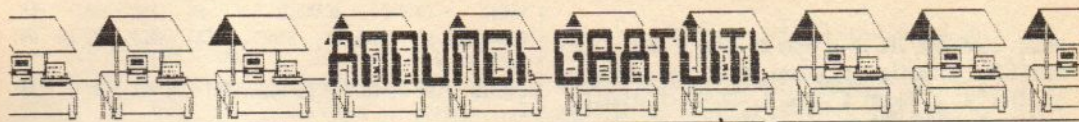
```

```

239 REM -----
-----***
240 POKECM,OC:POKECM+1,OC:GOSUB181:
241 NN=1:GOTO125
242 REM -----,***
*
243 FORI=0TO20:NEXT
244 POKECM,OC:POKECM+1,OC
245 FORI=0TO20:NEXT
246 GOTO216
247 DATA160,00,165,254,145,251,200,19
6,253,208,249,96,252,0,3,255
248 DATA0,3,3,128,7,57,128,7,35,128,7
,57,128,3,3,128,3,255,0,0,252,252,0
249 DATA3,255,0,3,3,128,7,207,128,7,2
07,128,7,207,128,3,3,128,3,255,0,0,25
2
250 DATA252,0,3,255,0,3,25,128,7,25,1
28,7,41,128,7,49,128,3,57,128,3,255,0
251 DATA0,252,252,0,1,131,0,3,49,128,
6,121,128,6,127,128,6,113,128,3,57
252 DATA128,3,131,0,0,252
253 DATA252,0,1,131,0,3,57,128,6,124,
128,6,124,128,6,124,128,3,57,128,3,13
1
254 DATA0,0,252
255 DATA0,126,0,1,255,128,7,255,224,3
1,255,248,63,255,252,63,255,252,127,2
55
256 DATA254,120,0,30,248,0,31,248,0,3
1,248,0,31,120,0,30,127,255,254,63,25
5
257 DATA252,63,255,252,31,255,248,7,2
55,224,1,255,128,0,126,0,0,0,0,0,0,0
0
258 DATA38,43,39,38,43,39,183,45,39,1
83,45,44,183,45,64,113,38,46,113,38,4
2
259 DATA38,43,32,38,43,36,38,43,47,64
,34,50,64,34,54,113,38,37,113,38,39
260 DATA113,38,51,64,34,57,83,32,53,1
68,25,72,204,28,54,83,32,51,64,34,62
261 DATA64,34,75

```





Tutti i lettori che desiderano comunicare con il mercato, possono approfittare della rubrica di piccoli annunci gratuiti, compilando e spedendo a: CO.STA EDIZIONI - VIA MAZZINI 128 - 50100 PISA il sottostante tagliando.

___ VENDO	___ COMPRO	___ SCAMBIO	___ INFORMO
___ HARDWARE	___ SOFTWARE	___ ALTRE	
NOME _____			
VIA _____		CITTA' _____	
C.A.P. _____	TELEFONO _____	ORARIO _____	

VENDO - SCAMBIO - PROGRAMMI per COMMODORE 64 su dischetti e cassette.

Scrivere a:

TAGLIACCOZZO ARIEL - Via Sabino, 16 - 00199 - ROMA

VENDO - SCAMBIO - PROGRAMMI per il COMMODORE 64 (oltre 470) sia su nastro sia su disco. Ultime novità per cassetta.

Scrivere o telefonare a:

MARANGONI ANDREA - Via Pausula, 72 - 62014 - CORRIDONIA
Tel. 0733/ 292416 - ORARIO: pasti.

VENDO - COMPRO - SCAMBIO - PROGRAMMI per il COMMODORE 64. Tra gli altri, un eccellente TOTOCALCIO in linguaggio macchina. Accetta qualsiasi tipo di condizionamenti. Scrivere o telefonare a:

STOLZI LIBERO - Via S.Maria, 1 - 53021 - ABBADIA S.S. (SIENA) - Tel. 0577/ 848117 - ORARIO: 13.00 - 17.00

SCAMBIO - VENDO programmi per COMMODORE 64, preferibilmente su disco - ANNUNCIO SEMPRE VALIDO - Inviatemi le vostre liste.

SICCARI ROBERTO - Via Matteotti, 178 - 18038 SANREMO
Tel. 0184/ 75687 - ORARIO: 15 -19

VENDO - VIC 20 + registratore + 7 libri d'informatica + 5 cassette di giochi di cui 2 cassette d'istruzioni. PREZZO L.200.000.

DE STEFANO CESARE - Via Napoli, 81 - 80053 - C/MARE DI STABIA - Tel. 081/ 8722007 - ORARIO: 20.00 - 22.00

Scambio programmi con tutta l'Italia. Scrivere o telefonare a: UMBERTO BENELLI - Viale XX Settembre 180 54031 - CARRARA-AVENZA (MASSA).

Tel. 0585/ 57145 - Telefonare dalle 13 alle 23.

Vendo COMMODORE VIC 20 (1 anno di vita) + registratore COMMODORE + tantissimi giochi fra i piu' validi + joystick a 2 fuochi a sole L.330.000 trattabili. Telefonare a: EDOARDO GILLONE 051/758026 ore pasti.

Vendo floppy disk COMMODORE 1541 per :VIC 20, C16, C64, Plus 4, nuovissimo a L.490.000.
GIANCARLO CARBONE - Via G.Papini,8 - 80046 S.Giorgio a Cremano (NA). - Tel. 081/7715274.

Cerco disperatamente il libro del sistema operativo del C16 / Plus 4. Solo se con commento in italiano o in inglese. Massima serietà. Telefonare o scrivere a: TOMMASO MAZZONI - Via S.Pietro,40 - 17024 Finale Ligure (SV). - Tel. 019/692332.

Cambio programmi per COMMODORE 64 sia su disco che su nastro. Telefonare o scrivere a:
Carlo Di Chiara Maggioli - via Carlo Dossi 15 - 00137 ROMA. - Tel. 06/8276440.

Scambio - Acquisto - Vendo programmi per il COMMODORE PC10 e COMMODORE PC20 ed IBM COMPATIBILI.
Scrivere a: Negri Giusy - Via Adone del Cima, 39E/12 55048 - TORRE DEL LAGO - (LUCCA).

COMPRO stampante per COMMODORE 64 MPS801 (meglio se 802) o affine, usata, in buono stato, preferibilmente con manuale d'uso.
Scrivere o telefonare a:
MONTANARI NICOLA - Via Genio, 9 - 40135 - BOLOGNA
Tel. 051/413453 ore dei pasti.

VENDO COMMODORE 64 piu' stampante MPS801, piu' registratore C2N, joystick e oltre 30 giochi a lire 1.000.000. Inoltre regalo un pacco da 2.000 fogli per stampante, alcuni volumi ed eventualmente anche televisore B/N. Scrivere o telefonare a:
TANFANI UGO - Via Monti, 1 - 20090 TREZZANO s/N - (MI)
Tel. 02/4455748.

VENDO - COMPRO - SCAMBIO SOFTWARE (PROGRAMMI) PER COMMODORE 64, UTILITY, GAMES, GESTIONALI, NE HO UN PACCO!

CAMBIO VIC 20 DOTATO DI ACCESSORI CON TRX PER I 144 MHZ.

Scrivere o telefonare a: Centi Luigi - Via C.Orientale - 67039 - SULMONA.
Telefono: 0864/ 52694 - ore 13.00 - 15.00 e dalle 21.00 in poi.

VENDO PROGRAMMI PER COMMODORE 64, dispongo di molti giochi ed utility solo pero' su cassetta.
ANNUNCIO SEMPRE VALIDO.

Scrivere a: Bonelli Stefano - Via g. di Vittorio, 7 - 53014 - MONTERONI D'ARABIA (SI).

VENDO VIDEO-GAMES PER COMMODORE 64, ultime novita' tra cui i favolosi SUMMER GAMES II e MISSIONE IMPOSSIBILE II a prezzi bassissimi.

Scrivere o telefonare a: Leone Marco - Via Giacomo Corradi, 13 - 00151 ROMA.
Telefono: 06/ 5341402 - ore pasti.

VENDO VIC 20 + REGISTRATORE + MANUALE VIC FORTH + MANUALE VIC REWALD + CARTUCCIA FORTH A SOLE LIRE 150.000 (TRATTABILI). VENDO ANCHE SEPARATAMENTE.
Scrivere o telefonare a: Caposio Massimiliano - Corso Tazzoli, 200 - 10137 - TORINO.
Telefono: 307481 - ore 17-21.

COMPRO stampante compatibile COMMODORE, possibilmente in buono stato, in cambio di 500 programmi e modico compenso.
Telefonare o scrivere a: Chiocci Giuseppe - Via Savelli, 8 - 06024 - GUBBIO (PERUGIA).
Telefono: 075/9271549 - ore pasti.

VENDO , SCAMBIO programmi di giochi per il COMMODORE 64, ne ho parecchi e cerco ragazzi della mia eta' per scambiarmi idee e sui vari programmi; io ho 13 anni. Ho un amico che possiede il COMMODORE 16, se volte scambiare programmi anche per questa macchina, scrivetemi o telefonatemi.
Bonati Francesco - Via B.Cremagnani,1 - 20059 - VIMERCATE - (MILANO).
Telefono: 039/ 669796 - ore 14-15 e ore 18-20.

Scambio programmi COMMODORE. Scrivere o telefonare a:
LEONARDO DE PALMA - Via Francesco Marinaccio, 7 - 71100
FOGGIA - Tel. 0881/ 79700 - Telefonare nel pomeriggio.

SCAMBIO PROGRAMMI COMMODORE. Scrivere o telefonare a:
ROBERTO OSELLADORE - Via Passo San Boldo, 35/2
30030 - FAVARO VENETO (VENEZIA).
Tel. 041/ 631106 - Telefonare dalle ore 12 alle ore
13.30 e dalle ore 17 alle ore 18.15.

Scambio programmi COMMODORE. Scrivere o telefonare a:
FRANCESCO DE COLLE - P.le Capo Linaro 11/23
00053 CIVITAVECCHIA (ROMA)
Tel. 0766/ 34171 - Telefonare ore pasti.

Scambio programmi. Scrivere o telefonare a:
GIANNI GAGGINO - Via Graglia 18
10136 - TORINO
Tel. 011/ 352830 - Telefonare dalle ore 12.30 alle ore
14.30 Oppure dalle ore 19.30 alle ore 21.30.

Scambio programmi. Scrivere o telefonare a:
LUCA LORENZINI - Via Lumumba, 11
41011 - CAMPOGALLIANO (MO)
Tel. 059/ 525861 - Telefonare dalle ore 14 alle ore
15.30 e dalle ore 18 alle ore 21.

Scambio programmi su computer della COMMODORE. Scrivere
o telefonare a:
MASSIMILIANO GRENANIN - Via XXV Aprile, 4
10036 - SETTIMO TORINESE (TO)
Tel. 011/ 8010392 - Telefonare dalle ore 18 alle ore
20.

VENDO PROGRAMMI per COMMODORE 64 per TOTOCALCIO e
TOTIP.
Totocomputer, Toto 13, Totocalcio, Totoman,
Totocompilatore a L.15.000 cad. su cassetta e a
L.18.000 su dischetto. Inoltre una completa gestione
condominiale prevista anche delle stampe a sole
L.15.000. 5 giochi a L.6.000 su cassetta ed a L.9.000
su dischetto. LISTATI GRATUITI!!! Inviare bollo.
Manfre' Nunzio - Via Fe'D'Ostiani, 22 - 25020 - OFFLAGA
(BRESCIA).
Tel. 030/ 979491 - Ore 14.30 - 16.30.

SCAMBIO programmi per il COMMODORE 64, dispongo di
parcchie delle ultime novita'. Inviatemi la Vostra
lista ed io Vi inviero' la mia.
Scrivere a: Mancini Mauro - Via Lucrino, 41 - 00199 -
ROMA.

VENDO - SOFTWARE - HARDWARE: Vendo per COMMODORE 64
programmi (2000 giochi ed utility su nastro e disco)
ed hardware (tasto reset, interfaccia doppio tape,
espansione memoria per VIC 20 ecc.). Prezzi veramente
interessanti.
Scrivere o telefonare a: Cottogni Gianni - Via
Strambino, 23 - 10010 - CARRONE (TORINO).
Telefono: 0125/ 712311 - ore 18.00 - 21.00.

CAMBIO ED EVENTUALMENTE VENDO GIOCHI ED UTILITES PER
COMMODORE 64 a L.2.000 - 5.000 cad.
Ne possiedo di veramente fantastici tra cui: PENGU,
DECATHLON, GHOST BUSTER, REACK DANCE, FLIPPER,
SIMULATORE DI VOLO ecc. Offro ed esigo la massima
serietà. Scrivere o telefonare a:
FRANCESCO TAGLIAPIETRA - Stradella dell'ospedale, 30 -
36100 - VICENZA.
Tel. 0444/ 37246 - Ore 18.00 - 20.00.

SICURAMENTE HO TUTTO QUELLO CHE ANDATE CERCANDO.
Il software piu' valido ed esclusivo che oggi sia
possibile trovare. Rispondo a tutti.
SCRIVERE a: CATONE DOMENICO - Via Viticella, 243 -
80010 - QUARTO - (NAPOLI).

VENDO SU DISCO/ CASSETTA 7 programmi per sfruttare in
maniera veramente completa le possibilita' musicali del
COMMODORE 64. Tutto a sole L.50.000. Vendo inoltre
sostanzioso programma per la progettazione ottica di
telescopi NEWTON- CASSEGRAIN, SCHMIDT, con
implementazione di tutti i comandi grafici H/R in L/M.
Scrivere o telefonare a:
COLLI MAURO - Via Bachelet, 3 - 66023 - FRANCAVILLA
(CH).
Tel. 085/ 817563 - Ore dei pasti.

FAVOLOSA ED UNICA OCCASIONE PER VENIRE IN POSSESSO DI
TANTISSIMI GIOCHI A PREZZI BASSISSIMI.

Le ultime favolose novità di quest'anno, importate direttamente dall'estero. Sconti mostruosi: 250 giochi a sole L.100.000. Spedizioni e supporti a Vostro carico. Sono sempre disponibile anche ad eventuali scambi. SCRIVERE O TELEFONARE A:
GIARDINA GIUSEPPE - C.so Umberto I, 134 - 70017 - PUTIGNANO (BARI).
Tel. 080 / 781281 - Ore 10.00 - 17.00

PER AUMENTARE LA MIA COLLEZIONE DI PROGRAMMI (OLTRE 1.000), cambio con chiunque software per il COMMODORE 64 su disco o cassetta. Dispongo delle ultimissime novità. Per scambio delle liste scrivere a:
SOFTWARE BANK ITALIA - Casella postale 28 - 22066 - MARIANO COMENSE - (COMO).
Tel. 031/ 747367 - Ore 15.30 - 19.00.

ATTENZIONE - ATTENZIONE - ATTENZIONE - ATTENZIONE

COMUNICATO A TUTTI I LETTORI DELLA RIVISTA:

INVIATE I VOSTRI MIGLIORI PROGRAMMI DI:
GIOCHI, UTILITA', PERSONALI, GESTIONALI; I VOSTRI TRUCCHI (per la rubrica TRUCCHI).
I MIGLIORI RICEVERANNO UN PREMIO E SARANNO PUBBLICATI NELLA RIVISTA.

PER POTER ESSERE PUBBLICATI, DOVRANNO ESSERE BEN DOCUMENTATI, NON COPIATI DA ALTRE RIVISTE DEL SETTORE, POSSIBILMENTE CARICATI SU DI UN SUPPORTO MAGNETICO (CASSETTA O DISCHETTO). PER BEN DOCUMENTATI, SI INTENDE CHE DOVRANNO ESSERE DESCRITTI I VARI BLOCCHI D'ISTRUZIONI, DI CUI SONO COMPOSTI ED IL MODO IN CUI DOVRANNO ESSERE UTILIZZATI, DAI LETTORI CON EVENTUALI TRUCCHI D'UTILIZZO.

ATTENDIAMO I VOSTRI LAVORI, CERTI CHE SARETE NUMEROSI.
UN BUON LAVORO A TUTTI.

COMMODORE DOSSIER

CODICE	QUANTITÀ	PREZZO TOTALE
TOTALE L.		

Nome _____

Cognome _____

Via _____ N° _____

Città _____

_____ CAP _____

Firma _____



PERSONAL COMPUTER **CLUB**

GIOCHI-CONCORSO

PROGRAMMI E TRUCCHI

PROVE HARDWARE E SOFTWARE

PROGETTI SOFTWARE

BIBLIOTECA PERSONALE

e le rubriche

TUTTOCOMMODORE

TUTTOSINCLAIR

TUTTOMSX

**OGNI MESE
IN EDICOLA
LIRE 3.000**